APPLIED COMPUTER SCIENCE

The Journal is a peer-reviewed, international, multidisciplinary journal covering a broad spectrum of topics of computer application in production engineering, technology, management and economy.

The main purpose of Applied Computer Science is to publish the results of cutting-edge research advancing the concepts, theories and implementation of novel solutions in computer technology. Papers presenting original research results related to applications of computer technology in production engineering, management, economy and technology are welcomed.

We welcome original papers written in English. The Journal also publishes technical briefs, discussions of previously published papers, book reviews, and editorials. Especially we welcome papers which deals with the problem of computer applications in such areas as:

- manufacturing,
- engineering,
- technology,
- designing,
- organization,
- management,
- economics,
- innovations,
- competitiveness,
- quality and costs.

The Journal is published quarterly and is indexed in: BazTech, Cabell's Directory, CNKI Scholar (China National Knowledge Infrastucture), ERIH PLUS, Index Copernicus, J-Gate, Google Scholar, TEMA Technik und Management.

Letters to the Editor-in-Chief or Editorial Secretary are highly encouraged.

CONTENTS

Applied Computer Science, vol. 14, no. 1, pp. 5–15
doi:10.23743/acs-2018-01

Submitted: 2018-02-20
Revised: 2018-03-05
Accepted: 2018-03-20

road lights intelligent system, Internet of Things, GPS localization, MQTT message transmission protocol

Sebastian CYGAN[*], *Barbara BOROWIK*[**], *Bohdan BOROWIK*[***]

STREET LIGHTS INTELLIGENT SYSTEM, BASED ON THE INTERNET OF THINGS CONCEPT

Abstract

The paper presents a project of a street lights intelligent system, which allows for public savings, as a result of more efficient roads and pavements lights management. The system would operate based on the current location of vehicles and pedestrians. Because of this no additional costs or devices are required in smartphones or modern vehicles to indicate its location, as smartphones, smartwatches and most of modern cars are equipped with the GPS modules. But each smart street lamp needs to be equipped with a little module that communicates with the central system and which controls the work of lighting.

1. INTRODUCTION

Internet of Things is a concept, which assumes the possibility of connection to global Internet network many devices. Their quantity and possibility of mutual communication often allow to create autonomous, intelligent systems, freeing a human being from many daily activities (Mitchell, 2007; Ashton, 2017). According to forecasts in 2020 around 50 billion devices used worldwide will belong to the IoT (Markowski, 2017). Such rapid development of this branch of IT and electronics is caused by the increased need for complex systems, which would allow autonomously manage the work of intelligent vehicles, houses or whole cities (Mitchell, 2007; Stawasz & Sikora-Fernandez, 2016).

[*] ELTE GPS, ul. Medyczna 13, 30-688 Kraków, sebastian.cygan91@gmail.com
[**] Politechnika Krakowska, Instytut Informatyki, ul. Warszawska 24, 31-155 Kraków, cnborowi@cyf-kr.edu.pl
[***] Akademia Techniczno-Humanistyczna, Katedra Elektrotechniki i Automatyki, Bielsko-Biała, ul. Willowa 2, 43-309 Bielsko-Biała; E-mail: bo@borowik.info

It has been calculated, that roads and pavements lighting accounts for up to 20% of electricity generated by cities (Miller, 2016).

Intelligent street lighting system could allow for a significant reduction in public electricity costs. It would operate based on the current location of vehicles and pedestrians. If there would be no objects near the street lamp, then this device (depending on the configuration) could be turned off or could have its lighting power reduced. For this purpose there could be used such devices that nearly every one of us already has: smartphones, smartwatches or modern cars, equipped with the GPS modules (Markowski, 2017). In addition, each smart street lamp will be equipped with a little module that communicates with the central system and which manages the work of lighting.

1.1. Related works

The subject of effective management of street lighting has been often the subject of research especially during the last years. For example the authors of (Drechny & Kolasa, 2016) in their intelligent algorithm have used artificial neural network. Controlling of a street lighting depends on the intensity of a traffic (data can be obtained from the supervision center system) and may use the information concerning weather conditions (forecasted and current). A street lighting system may operate and control lamps in a zero-one mode (ordinary lamps) or may adjust the degree of the light intensity (LED light bulbs). Another concept has been proposed by the LonWorks technology. As part of the research in 2014 the University of Science and Technology (AGH) in Krakow had launched a pilot system that had supervised the work of public lighting in one of its car parkings (Ożadowicz & Grela, 2014). Besides the whole range of parameters allowing for monitoring energy consumption and detecting potential threats to energy infrastructure the system was allowing for both manual and automatic tuning based on data that was obtained from sensors of ambient light and of temperature. The whole system provides bigger reliability and significant savings in energy consumption. Poland, in the context of modern technologies, is an attractive market for the implementation of intelligent street lighting.

1.2. The use of mobile devices in transmitting the location of objects

Nowadays, the majority of people are equipped with modern smartphones, which besides their standard capabilities provided in traditional telephones, they also offer very specialized functions: GPS location, possibility of internet connection and they are based on advanced operating systems, on which own software can be implemented. Because of this no additional costs or devices are required in smartphones or modern vehicles to indicate its location. Modern cars are more and more often equipped already in factories with GPS modules and with devices that allow them for network connectivity.

1.3. Devices allowing to manage the work of street lamps

An important function of street lamps is the necessity to exchange data with the central server (based on the GSM network). There exist also a possibility of a hardware integration with street lamps, so the control over these lamps work could be performed.

An interesting option on the market is the one based on a Raspberry Pi (Retrieved from https://www.raspberrypi.org). It is a miniature computer that allows for connecting any device via USB or an Ethernet interface. Thanks to this it is possible for this device to quickly connect to the mobile network, while its dedicated distribution of the Linux operating system allows to run multiple applications on it. The entire such a system is very energy efficient, because it requires only approximately 500 mA in its normal operating mode.

1.4. A protocol enabling mutual communication of devices

The basis of every system consisting of a large number of devices is their mutual communication. Many dedicated protocols have been developed for the needs of IoT, in which the most important is the speed of data transfer, the smallest communication overheads and the ability to encrypt the transmission. It was established that the communication medium between all devices considered in the implemented project would be GSM network, because of its universality and the lack of the need to build a dedicated infrastructure. From the many Internet of Things protocols, the authors have selected the MQTT protocol ("MQTT Protocol", 2017), which is popular as it is used for example in Facebook Messenger ("Tops of 2016: digital", 2016).

2. THE PROPOSED SYSTEM

The main assumption of the project is the energy-saving mode, for which the period is searched, when the light emitted by street lamps can be diminished or reduced. By definition, it will be during late night hours, when the total traffic is negligible and there is no need to constantly illuminate roads and sidewalks. At these periods of time the work of lamps will depend on the proximity of vehicles and pedestrians moving around the city, i.e. the light will be dynamically adjusted to the speed and direction of objects' moving. Figure 1 presents the example of a graph showing the time function and the change of the street lamp working modes.

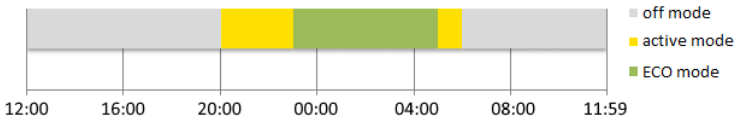

Fig. 1. Examples of street lamp work stages on the time function graph

Except for the periods of validity of the energy-saving mode (when the light emitted by the lanterns will be reduced) the system will only record data on the location of facilities and verify the efficiency of streetlights. Such data may allow to carry out a significant analysis of the traffic load at particular times of the day and they can enable quick response to possible failures of the street lighting network. In the energy-saving mode, by default all lanterns will work in low-power mode or completely blank (depending on the configuration).

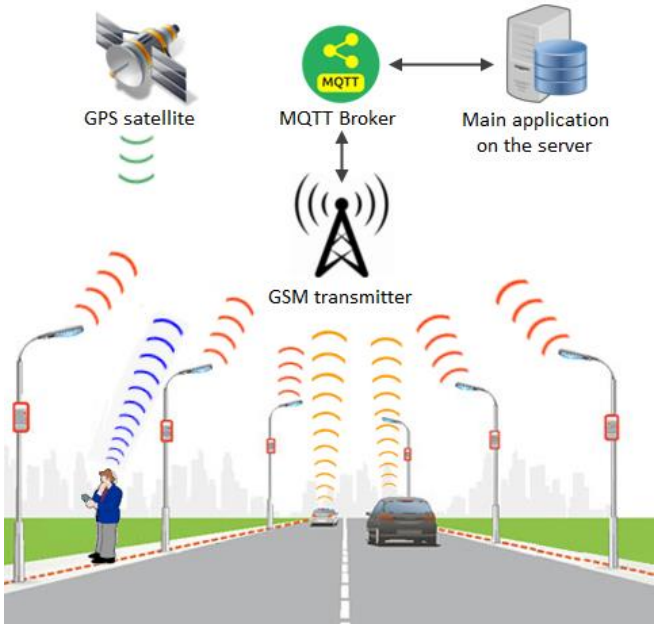

Fig. 2. Scheme of data exchange between objects in the system and the server – in the GPS technology users' and vehicles' mobile devices transmit their current locations to the server, and this allows for the management of street lamps by a central application

So each vehicle and every pedestrian moving around the city should have running a mobile application, which transmits the information about their current location to the central server. Based on these data, the main application will determine sequences of street lamps which will be successively starting their work, so to ensure the continuous lighting of roads and/or pavements for moving objects.

If there would be no vehicles or pedestrians in the given area, messages would be sent to the street lamp, with the instructions to re-enter the energy-saving mode. The graphical concept of this system has been shown in Figure 2. The system project used the MongoDB database.

2.1. Dependencies between objects in the proposed system

The basis of the algorithm controlling the work of street lamps is obtaining the information about the location of individual objects (vehicles and pedestrians) moving around the area controlled by the system. Each of the street lamp in this area is characterized by important parameters used to identify it in the context of surrounding moving objects. The first such a parameter is the radius (expressed in meters). Within its range, objects are clearly classified as located within the area of a given street lamp. In addition to the obvious scenario, in which a vehicle's or a pedestrian's coordinates are evidently within the range of this street lamp (what is the case presented in Figure 3), it also may happen that the coordinates of two objects will cause omission of obtaining the information about the objects being found near this street lamp (this has been presented in Figure 4).

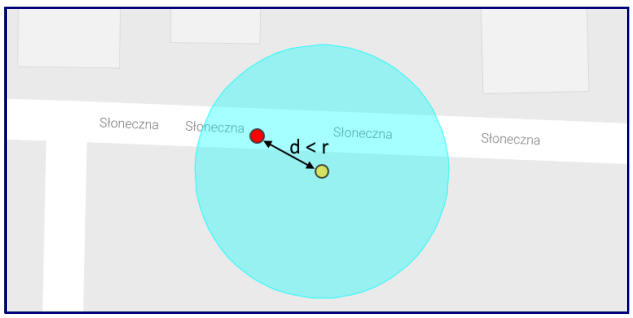

Fig. 3. The situation, in which the object is within the range of a street lamp (the distance is smaller than the defined radius of the visibility range of a given street lamp)

In this case, the algorithm performs the following operation: a straight line is determined between the two data points. Then, from the coordinates of the central point of this street lamp is determined a perpendicular line to line between the two points of the mentioned earlier object. The point of the intersection of these two lines determines the closest place to the street lamp, the distance of which is checked by the program. If it is smaller than the radius of the lamp, then the algorithm classifies the presence of the object as within its range.

The second parameter, that is used to identify the given street lamp in the context of surrounding moving objects is its street lamps group number and its sequential number within a given group.

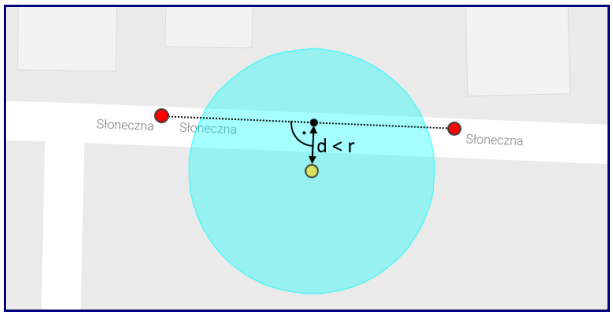

Fig. 4. The situation, in which the points of occurrence of an object have the location data that is not within the direct range of the given street lamp – however, based on the appropriate mathematical calculations, the algorithm recognizes the presence of an object within the area of the street lamp

These parameters serve to find out in which direction a given object is moving. It is particularly important when a given vehicle is traveling on a two-way road, in which street lamps are on both sides of the road. Based on these parameters, the algorithm is able to activate only a part of the street lamps for the needed a light road lane, consistent with the direction of the object movement, as it has been presented in Figure 5.

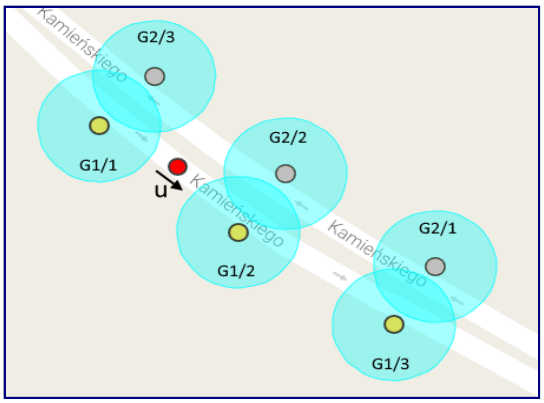

Fig. 5. The situation, in which there are street lamps on both sides of a two-directional road – the algorithm will activate only the correct set of available street lamps on one side of the road, based on determining a given lamp group number and its sequential number within a given group

2.2. The algorithm controlling the work of street lamps

The algorithm that controls the work of street lamps operates only during the energy-saving mode, when (depending on its configuration) the light of street lamps is dimmed or lamps shine with less intense light.

A simplified block diagram of the algorithm activating street lamps is shown in Figure 6.

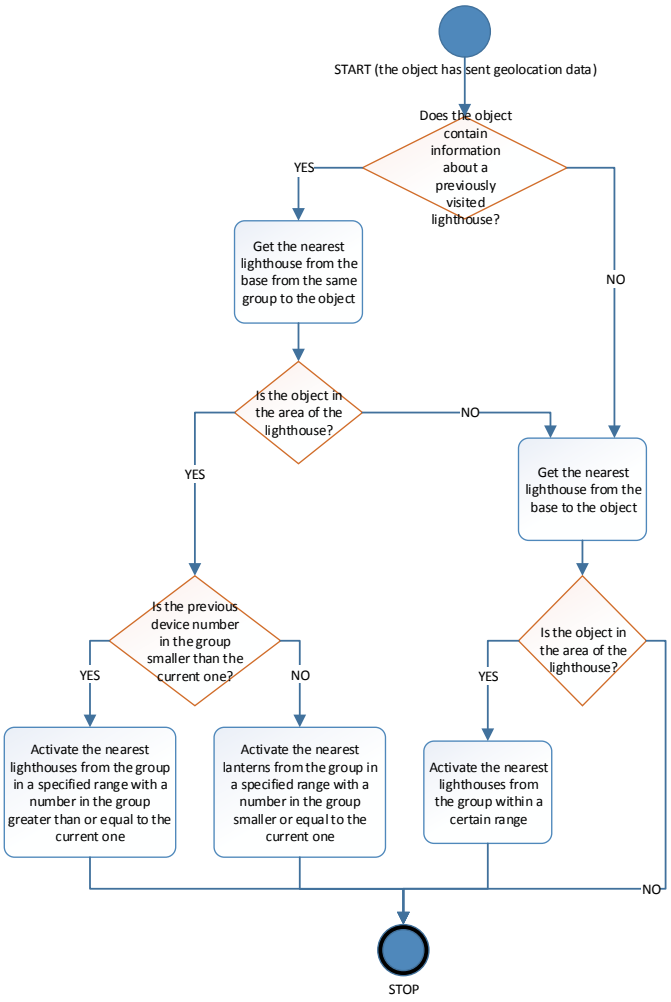

Fig. 6. A block diagram of an algorithm that activates street lamps, after receiving data about an object moving towards this lamp – in the case of pedestrians, it is not necessary to check the state of the previous lamp number within the group of lamps

Because each street lamp may have defined different hours of energy-saving mode (for example, the lamps located in the middle of a city may have a requirement of a longer work in a normal mode), the global ECO mode for the proposed system is determined by algorithm, as:

– start – the street lamp with the first minute of the first hour of the activation of the energy-saving mode,
– end – the street lamp with the last hour of its activation in the energy-saving mode or sun dawn, if it occurs earlier.

The scheme of the logical connections between street lamps and their groups is shown in Figure 7.

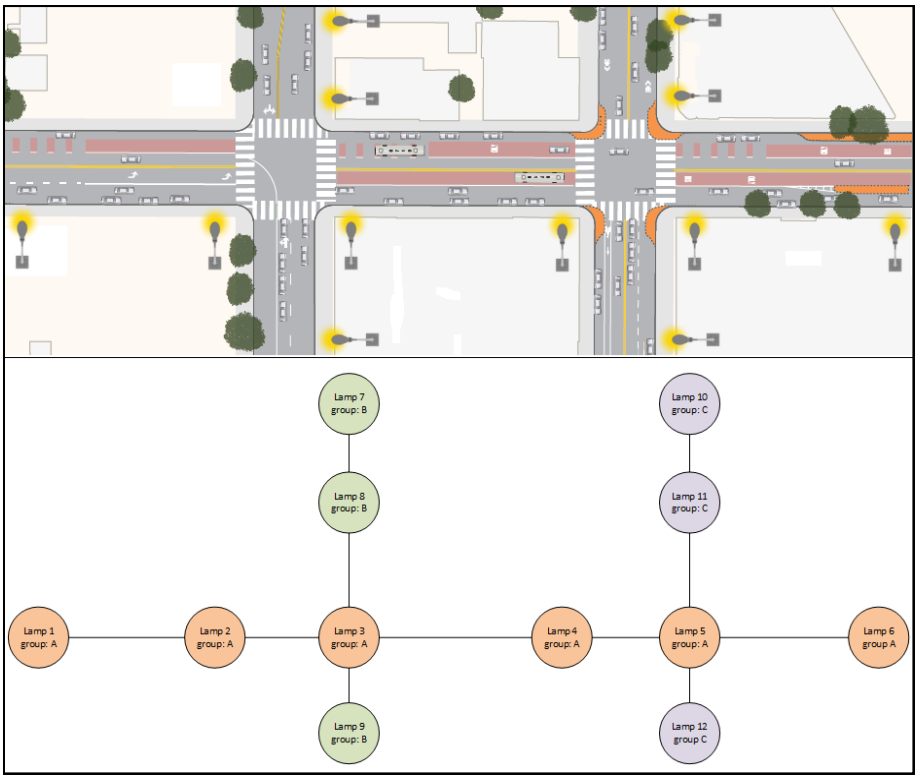

Fig. 7. An example of a mapping of street lamps networks to the form of logically connected objects, characterized by belonging to appropriate groups (corresponding to individual streets)

Assuming the activation of the algorithm one minute earlier than the minimum duration of the ECO mode is necessary to prevent the situation, when exactly at the moment of its inception the given place will go into darkness, and only the next data obtained from mobile devices will cause the street lamp to be reactivated. In turn, in summer time, a situation may occur, when the sunrise will come earlier than the morning hours of the energy-saving mode, and the operation of the algorithm should end, recognizing the sunrise. The additional

necessary functionality that this algorithm has it is to cause diminishing these street lamps, near which there have been no more vehicles or pedestrians. The solution has been solved in the following way: in the central application there is a function called cyclically (by default every 10 seconds), which for previously activated street lamps checks the following relationships:

- is the current time greater than the parameter declared for this lamp, which defines the minimum delay value that is needed to diminish the device after its previous activation;
- whether within an immediate vicinity of this lamp exist any other active objects;
- are there any other objects in its bigger neighborhood (around 1000 m by default) that move towards this given lamp (here is taken into account the azimuth parameter of the last data obtained from the given vehicle).

In the case that the program would determine the logical value *true* for each of the above conditions, then such a device would return to the energy-saving mode. The above assumptions make it necessary to set the additional flag for each lamp, defining the possibility of switching a given device into energy-saving mode by the algorithm. In a consequence, the system will not reduce the intensity of light of these lamps, for which the current time is not included in the duration of the ECO mode. Major difference in actions taken by the algorithm depends on whether a particular data obtained from a mobile device has been sent by a vehicle or by a pedestrian. It was assumed that the speed of the mo-vement of people is so small that there is no need to check the direction, in which they are heading. In this case, the algorithm works as follows: it determines the activation of a street lamp within the same group of devices, in which the person was previously located (if such an information is available). A distance of the activated elements of the lighting network from the pedestrian location - it is a parameter, configurable by the administrator, and by default it has value of 300 m in each direction. In the case of vehicles it is important to know, in which direction the given object moves. Having this information the system is able to activate successive lamps in the appropriate sequence, consistent with the direction that the given vehicle moves.

Presence of an object within the range of a lamp causes switching on the devices within a given lamp group and within the distance calculated from the following formula:

$$d = \frac{v}{\alpha \cdot 0{,}1}, \tag{1}$$

where: d – is a distance from a point [m],
v – is a given vehicle movement velocity [km/h],
α – is a coefficient, defined by the program [unit] (by default equals 1).

The lower limit of the determined distance by default equals 300 m, while the upper limit equals 1000 m. Because of this, for fast-moving vehicles the algorithm activates street lamps from a greater distance, what is to ensure adequate lighting of the proper part of a road.

Next, the system checks the following sequential device number within the group. If the previously visited lamp was the one with the number smaller than the current one, the street lamp with a bigger number or equal to the current device number in the group is activated. The algorithm works analogously in the opposite direction. If the vehicle does not store the information about the previously visited street lamp, then a default value (set up within the program) equal to 250 m is assumed in every direction within the same device group.

The above assumptions require that the following attributes need to be stored by individual elements of the collection within the system:

- vehicles and pedestrians – must store an identifier of the previously visited street lamp;
- a street lamp must store the number of its lamp group and the next device number within this group.

Additionally, each vehicle and every pedestrian stores a parameter value of a time stamp of the previously visited device. The central application continuously periodically calls a function that removes the reference to the previous lamp numbers if the object for a certain time (by default 1 minute) has not registered its presence within one of the lamps with a given group.

This prevents from cases in which objects that do not send its location for a longer period of time will be considered from the point of view of the previously activated lamps, where this information would usually no longer be reflected in reality.

3. CONCLUSIONS

The presented project of the system would certainly allow for the significant energy savings, taking into account all-night lighting of streets and pavements. The concept of the proposed system allows for dynamic adaptation of lamps to the current location of objects moving around the city, what ensures high efficiency of energy consumption. In recent years, these types of pro-ecological projects have been very popular as well as are willingly financed by various institutions (“Polska rynkiem rozwojowym dla inteligentnego oświetlenia ulicznego”, 2016). In the proposed system the costs of extra instrumentation of objects moving around the area are eliminated through the use of commonly used mobile devices. The project could in the future become a part of Smart Cities.

REFERENCES

Ashton, K., (2017, October 7). That 'Internet of Things' Thing. *RFID Journal*. Retrieved from http://www.rfidjournal.com/articles/view?4986

Drechny, M., & Kolasa, M. (2016). Sztuczna sieć neuronowa wspomagająca sterowanie oświetleniem ulicznym. *Rynek Energii, 2*, 90–95.

Markowski, A., (2017, May 25). *Niezbyt jasno pod latarnią*. Retrieved from http://www.forumsamorzadowe.pl/artykuly/artykul/209,0,45,0,niezbyt_jasno_pod_latarnia.html

Miller, M. (2016). *Internet Rzeczy. Jak inteligentne telewizory, samochody, domy i miasta zmieniają świat*. Warszawa: PWN.

Mitchell, W. (2007). Intelligent cities. *e-Journal on the Knowledge Society*. Retrieved from http://www.uoc.edu/uocpapers/5/dt/eng/mitchell.pdf

MQTT Protocol. (n.d.). Retrieved October 7, 2017, from http://mqtt.org

Ożadowicz, A., & Grela, J. (2014). Street Lighting – nowoczesne oświetlenie przestrzeni publicznych. Automatyka budynkowa w infrastrukturze inteligentnych miast – Smart Cities. *Napędy i sterowani, 16*(6), 104–108.

Polska rynkiem rozwojowym dla inteligentnego oświetlenia ulicznego. (n.d.). Retrieved October 7, 2017, from http://lighting.pl/Wydarzenia-branzowe/raporty-i-analizy/Polska-rynkiem-rozwojowym-dla-inteligentnego-oswietlenia-ulicznego

Project Raspberry Pi. (n.d.). Retrieved October 7, 2017, from https://www.raspberrypi.org

Stawasz, D., & Sikora-Fernandez, D. (2016). *Koncepcja smart city na tle procesów i uwarunkowań rozwoju współczesnych miast.* Łódź: Wydawnictwo Uniwersytetu Łódzkiego.

Tops of 2016: digital. (n.d.). Retrieved October 7, 2017, from http://www.nielsen.com/us/en/insights/news/2016/tops-of-2016-digital.html

Applied Computer Science, vol. 14, no. 1, pp. 16–26
doi:10.23743/acs-2018-02

Submitted: 2018-02-01
Revised: 2018-03-05
Accepted: 2018-03-22

numerical analysis, CFD, drag coefficient, motorcycle helmet

Zbigniew CZYŻ[*], *Paweł KARPIŃSKI*[**], *Tacetdin SEVDIM*[***]

NUMERICAL ANALYSIS OF THE DRAG COEFFICIENT OF A MOTORCYCLE HELMET

Abstract

The paper discusses a numerical investigation, using a CFD tool, ANSYS FLUENT, of drag acting on a motorcycle helmet. The simulations were performed on a model of a helmet downloaded from a free CAD model library. A solid model enabled us to generate a mesh, to define boundary conditions and to specify a model of turbulence. Accordingly, the values of forces acting on individual sections of the helmet were obtained and the coefficients of aerodynamic drag were calculated. The test results can be used to optimize the shape of the existing motorcycle helmet construction and to study the impact of generated drag forces on reaction forces affecting a motorcyclist's body.

1. INTRODUCTION

Motorcycle helmets, unlike cycling helmets, are of a complex design, have a protective face visor and are elongated enough to protect middle and lower head sections. Rice et al. (2016) state that the motorcycle helmet protects both the head and neck from injuries. Weight of such helmets is not as important

[*] Department of Thermodynamics, Fluid Mechanics and Aviation Propulsion Systems, Faculty of Mechanical Engineering, Lublin University of Technology, Nadbystrzycka 36, 20-618 Lublin, Poland, +48 81 538 47 64, z.czyz@pollub.pl
[**] Department of Thermodynamics, Fluid Mechanics and Aviation Propulsion Systems, Faculty of Mechanical Engineering, Lublin University of Technology, Nadbystrzycka 36, 20-618 Lublin, Poland, +48 81 538 47 64, pawel.karpinski@pollub.edu.pl
[***] Department of Mechanical Engineering, Faculty of Engineering and Natural Sciences, Altinbas University, Mahmutbey Mah, Mahmutbey Dilmenler Cad 26, 34218 Bagcılar/Istanbul, Turkey, tacetdin.sevdim@kemerburgazuniv.onmicrosoft.com

as that of cycling helmets. The key aspect is the best possible head protection in a crash. The helmet should protect not only against open wounds, but also significant acceleration – deceleration (*United Nations Economic Commission for Europe*, 2016). Helmet mechanical strength is regulated by the United Nations Economic Commission for Europe 22.05 Helmet Standard and the Federal Motor Vehicle Safety Standard No. 218 by the U.S. Department of Transportation. The relationship between a motorcycle helmet design and safety is comprehensively investigated by Fernandes and Alves de Sousa (2013).

Helmet strength and a helmet ability to fast absorb energy should be also accompanied by efficient helmet aerodynamics. The resulting drag should be as small as possible to have as reduced an impact on the human body, especially the neck, trunk and arms as possible. Drag can be reduced by changing the helmet shape, i.e. reducing the friction coefficient, or by reducing the front surface. The front surface cannot by freely modified due to design requirements as the helmet needs to be thick enough to efficiently protect the entire head surface. Reshaping the helmet, therefore, is the main method to reduce the emerging drag force while driving. Today's manufacturing technologies allow for even very sophisticated shapes. Typical production methods include material lamination or thermo-plastic processing. A rounded shape improves not only helmet aerodynamics, but also helmet strength (no indentations causing a locally accumulated tension).

One of the methods to aerodynamically examine the motorcycle helmet is a CFD-based numerical study which enables the distribution of velocity and pressure on the surface and around the research object. It is also possible to observe streamlines and to calculate drag coefficients. This method can be supported by experimental tests of the model, conducted in real conditions or in a wind tunnel.

Many research papers on the aerodynamic analysis of crash helmets focus on numerical testing of helmets for cyclists. Such structures have a streamlined shape (similar to the shape of a drop), cover only the upper part of the head, may have ventilation holes, and at the same time have a small mass. In addition, usually the air velocity flowing around the cycling helmet is less than the air speed flowing around the motorcycle helmet. Brownlie et al. (2010) present the results of aerodynamic drag tests for 12 different cycling helmets for different yaw angles. In the publication wrote by Beaumont, Taiara, Polidori, Trenchard and Grappe (2018) numerical tests of aerodynamic drag of various cycling helmets were made depending on the different position of the head in relation to the incoming air. With the strongly profiled shape of the helmet, the angle of the head position significantly influences the drag. This is due to the increase in the area of the frontal surface. This is confirmed by the research conducted by Alam, Chowdhurya, Zhi Weia, Mustarya and Zimmerb (2014), where the influence of ventilating holes placed on the surface of the helmet on the generated aerodynamic drag was examined.

In addition to the research of aerodynamics of crash helmets, there are publiccations investigating the aerodynamics of the entire driver's silhouette. Brownlie et al. (2009) show the impact of the cyclist's apparel on the generated aerodynamic drag. It is common knowledge that moving a two-wheeled vehicle behind the previous vehicle results in a reduction in the drag force. In the study by Blocken and Toparlar (2015), the influence of a car moving behind a cyclist on the drag force acting on it was examined. Numerical studies were compared with wind tunnel tests. It was found that a significant reduction in resistance (by several percent) occurs when the distance of the vehicle from the rider does not exceed 5 m. A similar research was carried out by Blocken, Toparlar and Andrianne (2016) where the car was replaced by a motorcycle. Blocken, Defraeye, Koninckx, Carmelit and Hespel (2013) also studied the aerodynamic drag generated by three cyclists moving behind them.

The aim of this work was to examine the drag coefficient of a motorcycle helmet, taking into account the interference drag generated by the human body. For this purpose, numerical calculations for different air flow velocities were made based on the prepared solid model of the test object. As a result of the calculations, the values of forces on individual elements of the test object were obtained. The surface of the elements necessary to calculate the resistance coefficient was measured using the CAD software.

2. RESEARCH OBJECT

The research object is a model of a motorcycle helmet downloaded from a free GrabCAD model library [grabcad.com] connected with a CATIA V5 model of a human trunk with arms and a neck. Connecting the research object with a part of the human body was to show real conditions of the airflow while riding a motorcycle. Such calculations will certainly include the geometry of the motorcycle. This approach, however, depends much on a type of motorcycle so the geometric model selected for our first calculations is shown in Figure 1.

The position of the model in regard to the global coordinate system is shown in Figure 2. The following axis designations have been adopted: x – longitudinal axis (in line with the direction of the velocity vector), y – transverse axis, z – vertical axis. The dimensions of the model are: 328 x 560 x 555 mm (length x width x height). The research object was placed in a computational domain with dimensions 5328 x 4560 x 3555 mm (length x width x height). A velocity inlet and pressure outlet were defined on it. The test object was defined as a wall element. The direction of the velocity vector of the incoming air is the same as the direction of the x-axis.

For such a model, a mesh of 1,522,977 Tet4- and Wed6-type cells, see Figure 1, was created. The model is segmented into basic elements such as a visor, a helmet body and a human body with a neck. Turbulence model k–ω SST, combining the

advantages of the k–ε and k–ω models, is applied in our simulation. This type of turbulence model perfectly maps the wall turbulent flow and still maps free flow turbulence. SST (Shear Stress Transport) means that the values of major tensions in the flow are limited. The calculation parameters selected in the solver are pressure-based and steady state.

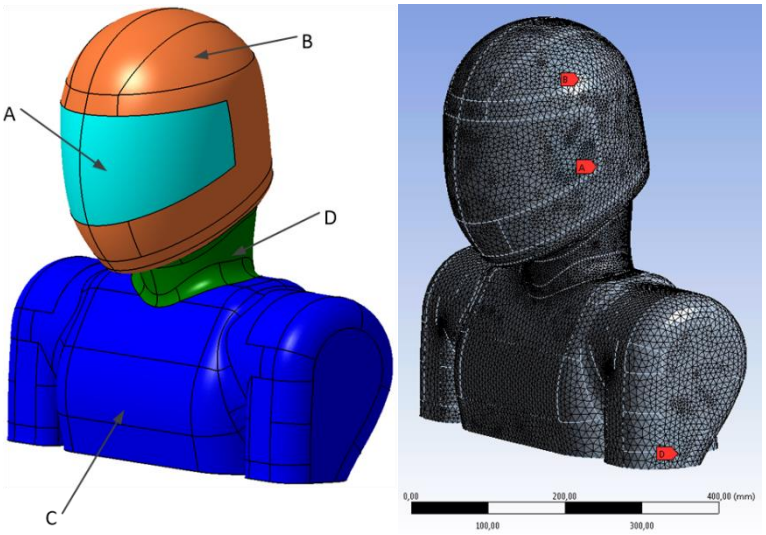

Fig. 1. The solid model of the test object divided into wall elements (visor A, helmet B, body C, neck D) and the view of the generated calculation grid model

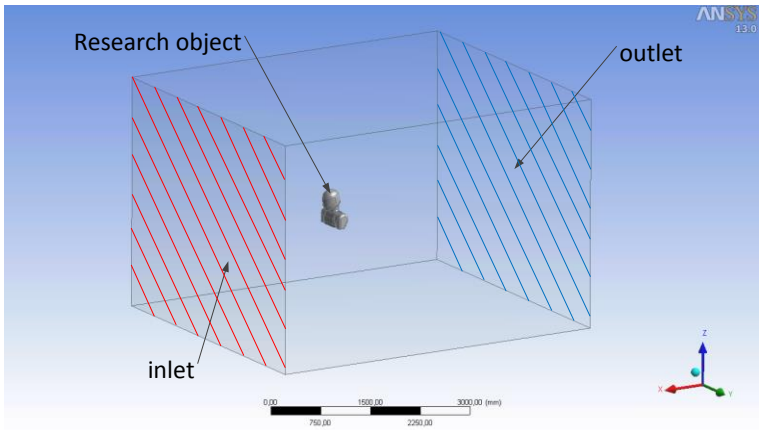

Fig. 2. The research object in the research area with the boundary conditions – isometric projection

The working agent is air defined as an ideal gas of 15°C and a viscosity of 1.7894·10-5 kg/(ms). The phenomenon of turbulence has been characterized by two parameters: intensity and length scale. It has been assumed that the intensity of turbulence is 5%, while the length scale at the inlet and outlet of the measurement area is 0.81 m. The length scale refers to the size of the turbulence in relation to the size of the duct.

For the model prepared in this way, a series of simulations were carried out for different velocities of the flowing air. Measuring points accepted for the calculation are: v = 5.56 m/s, 13.89 m/s, 27.78 m/s, 41.67 m/s.

3. RESULTS

As a result of the numerical analysis, pressure and velocity distributions were obtained on the tested motorcycle helmet for four calculation cases. Figures 3–5 present the examples of pressure contours on the surface of the research object and in the plane of symmetry for the velocity of the flowing air equal to 13.89 m/s, 27.78 m/s, 41.67 m/s, respectively. Figures 6–7 show the contours of velocity in the plane of symmetry for the velocity of flowing air equal to 5.56 m/s, 13.89 m/s, 27.78 m/s, 41.67 m/s, respectively. The drag forces generated on the individual helmet sections for varied airflow speeds are given in Table 1.

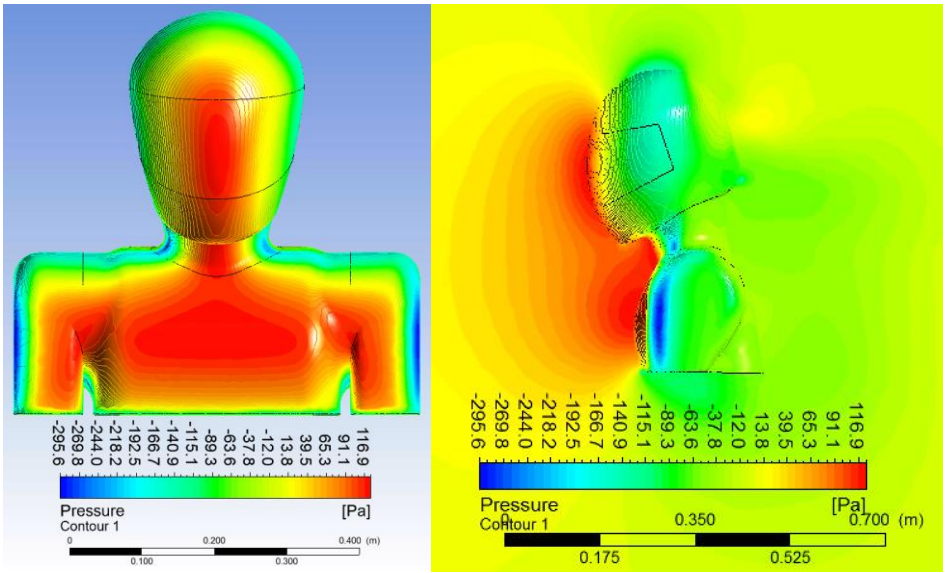

Fig. 3. The pressure contours on the surface of the research object and in the plane of symmetry for the velocity of the flowing air v equal to v = 13.89 m/s

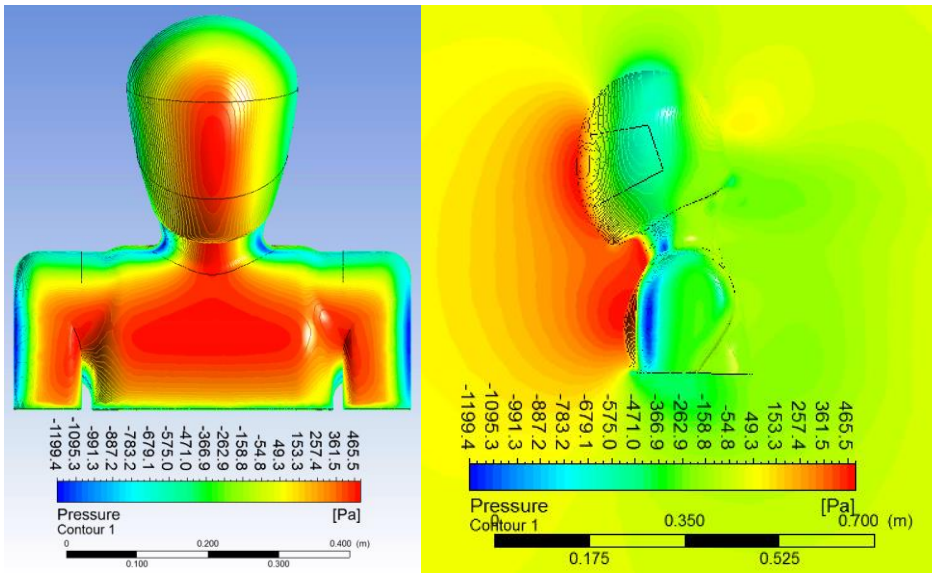

Fig. 4. The pressure contours on the surface of the research object and in the plane of symmetry for the velocity of the flowing air *v* equal to *v* = 27.78 m/s

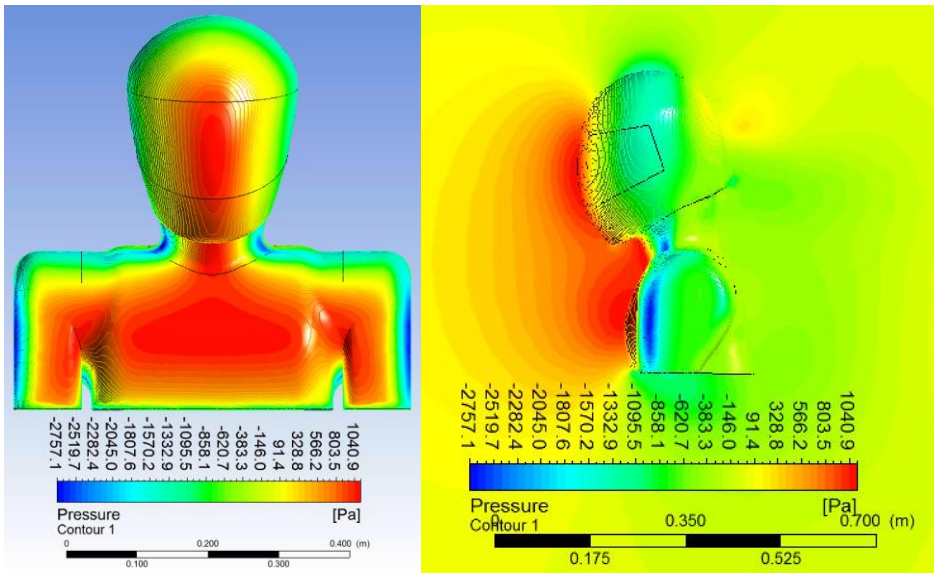

Fig. 5. The pressure contours on the surface of the research object and in the plane of symmetry for the velocity of the flowing air *v* equal to *v* = 41.67 m/s

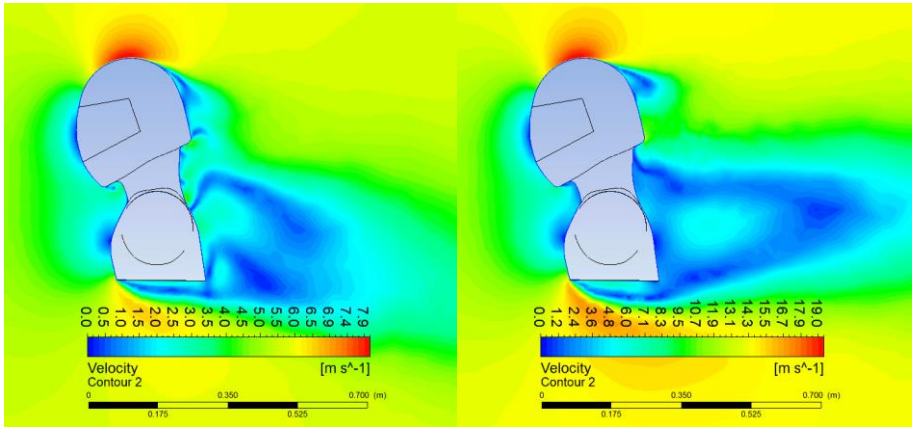

Fig. 7. The velocity contours in the plane of symmetry for velocity of flowing air *v* equal to 5.56 m/s (left) and for 13.89 m/s (right)

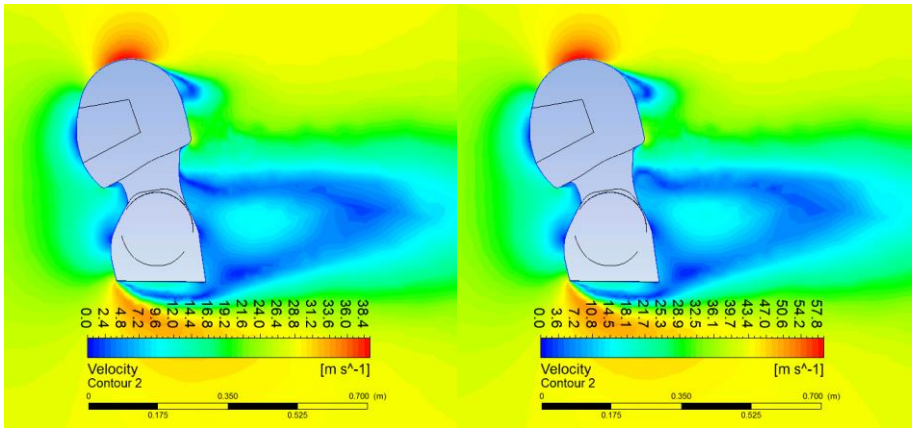

Fig. 8. The velocity contours in the plane of symmetry for velocity of flowing air *v* equal to 27.78 m/s (left) and for 41.67 m/s (right)

Tab. 1. Generated drag force for the research object

Case	Velocity *v* (m/s)	Drag force (N)				
		Visor	Helmet	Neck	Body	Total
#	0	0	0	0	0	0
1	5.56	0.155	0.547	0.209	1.980	2.890
2	13.89	1.180	2.833	0.962	10.726	15.701
3	27.78	4.582	10.558	3.501	42.431	61.0718
4	41.67	10.069	23.047	7.658	96.813	137.583

The front surfaces of the individual sections of the research object are measured in the CATIA V5, see Table 2. These values enabled us to calculate the value of the drag coefficient generated on the each of the sections of the research object in line with model (1), and the results are given in Table 3.

Tab. 2. Surface area of the individual elements of the research object

Element	Surface area S (m^2)
Visor	0.028187
Helmet	0.033473
Neck	0.006631
Body	0.127460

$$C_{xi} = \frac{P_{xi}}{0{,}5 \cdot \rho \cdot v^2 \cdot S_i} \tag{1}$$

where: P_{xi} – aerodynamic drag force on the i element,
ρ – air density,
v – velocity of flowing air,
S_i – frontal area of the i element.

Tab. 3. Calculated drag force coefficient for the research object

Case	Velocity v (m/s)	Drag force coefficient C_x (–)			
		Visor	Helmet	Neck	Body
#	0	–	–	–	–
1	5.56	0.291	0.864	1.667	0.821
2	13.89	0.354	0.716	1.227	0.712
3	27.78	0.344	0.667	1.117	0.704
4	41.67	0.336	0.647	1.086	0.714

The column charts in Figure 9 depict the correlation between the drag generated on the individual helmet sections and the airflow speeds.

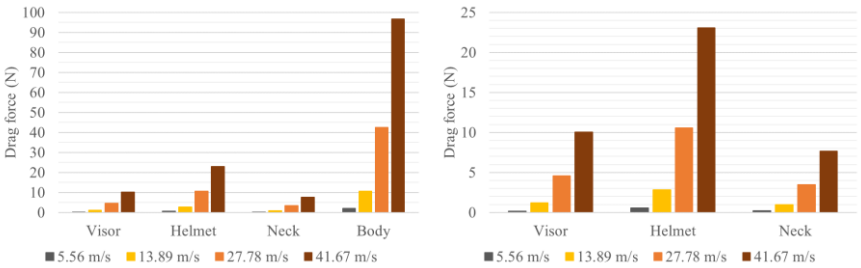

Fig. 9. Drag force on all of the simulated helmet sections (left) and the drag force for the selected helmet sections versus the flowing air velocity (right)

The total drag force as a function of airflow speed was plotted from the calculations. The plotted points enabled a 2nd order polynomial trendline, see Figure 10, according to equation (2):

$$P_x = 0.0061v^2 + 0.0033v + 0.189 \tag{2}$$

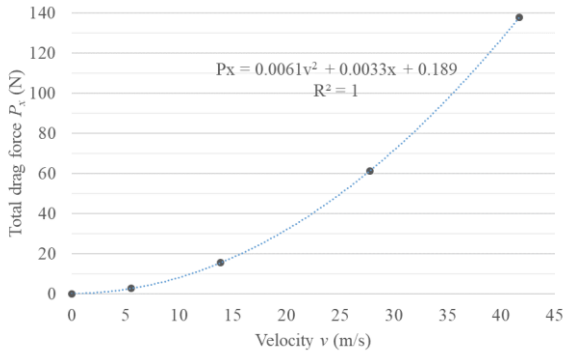

Fig. 10. The total drag force as a function of the velocity of the flowing air

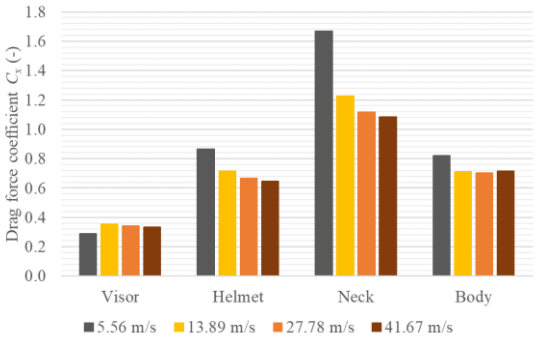

Fig. 11. Drag force coefficients for the individual elements of the research object depending on the velocity of flowing air

4. DISCUSSION AND CONCLUSIONS

Our calculations enabled us to claim that an accumulated pressure is the central sections of the visor and the trunk. The maximum value of overpressure for the airflow speed equal to 5.56 m/s is 21.39 Pa and increases with increasing speeds to reach 125.45 Pa, 500.2 Pa, 1120.05 Pa, respectively. The vacuum occurs on the rounded helmet edges, the neck edges and the shoulders as a result of the air speeding up at a rounded shape, which results in a decreased pressure (in line with Bernoulli's principle). The absolute value of the vacuum generated at the helmet edges is about 28 Pa at 5.56 m/s and is increasing with increasing speed to reach 146 Pa, 589, 1338 Pa, respectively. The highest vacuum occurs on the shoulder. The total drag force varies according to the square of the airflow speed. Its value for the airflow velocity of 41.67 m/s was equal to 137.6 N. This force is transferred to the arms and legs of the motorcyclist in the form of reaction forces. The largest value of the drag force component is generated on the motorcyclist's body. It can be reduced by tilting his body forward and using a motorcycle handlebars cover. Other elements (visor, helmet, neck) produce several times less drag. The drag force generated on the helmet's casing is twice as large as the force on the visor.

Based on the research performed, it was found that in the future it is possible to carry out extended aerodynamic analysis taking into account the additional geometry of the cover on the motorcycle handlebars for the different angle of the motorcyclist's body. This would allow us to examine the loads occurring on individual parts of the motorcyclist's body resulting from the impact of aerodynamic forces.

REFERENCES

Alam, F., Chowdhurya, H., Zhi Weia, H., Mustarya, I., & Zimmerb, G. (2014). Aerodynamics of ribbed bicycle racing helmets. *Procedia Engineering, 72*, 691–696. doi:10.1016/j.proeng.2014.06.117

Beaumont, F., Taiara, R., Polidori, G., Trenchard, H., & Grappe, F. (2018). Aerodynamic study of time-trial helmets in cycling racing using CFD analysis. *Journal of Biomechanics, 67*, 1–8. doi:10.1016/j.jbiomech.2017.10.042

Blocken, B., & Toparlar, Y. (2015). A following car influences cyclist drag: CFD simulations and wind tunnel measurements. *Journal of Wind Engineering and Industrial Aerodynamics*, *145*, 178–186. doi:10.1016/j.jweia.2015.06.015

Blocken, B., Defraeye, T., Koninckx, E., Carmeliet, J., & Hespel, P. (2013). CFD simulations of the aerodynamic drag of two drafting cyclists. *Computers & Fluids, 71*, 435–445. doi:10.1016/j.compfluid.2012.11.012

Blocken, B., Toparlar, Y., & Andrianne, T. (2016). Aerodynamic benefit for a cyclist by a following motorcycle. *Journal of Wind Engineering and Industrial Aerodynamics*, *155*, 1–10. doi:10.1016/j.jweia.2016.04.008

Brownlie, L., Kyle, C., Carbo, J., Demarest, N., & Harber, E. (2009). Streamlining the time trial apparel of cyclists: the Nike Swift Spin project. *Sports Technology, 2*, 53–60. doi:10.1002/jst.12

Brownlie, L., Ostafichuk, P., Tews, E., Muller, H., Briggs, E., & Franks, K. (2010). The wind-averaged aerodynamic drag of competitive time trial cycling helmets. *Procedia Engineering, 2*, 2419–2424. doi:10.1016/j.proeng.2010.04.009

Fernandes, F. A., & Alves de Sousa, R. J. (2013). Motorcycle helmets – a state of the art review. *Accident Analysis & Prevention, 56*, 1-21. doi:10.1016/j.aap.2013.03.011

Rice, T. M., Troszak, L., Ouellet, J. V., Erhardt, T., Smith, G. S., & Tsai, B.W. (2016). Motorcycle helmet use and the risk of head, neck, and fatal injury: Revisiting the Hurt Study. *Accident Analysis and Prevention, 91*, 200–207. doi:10.1016/j.aap.2016.03.002

United Nations Economic Commission for Europe. (2016). *The United Nations Motorcycle Helmet Study, United Nations*, New York and Geneva. doi:10.18356/82cd1e4b-en

Applied Computer Science, vol. 14, no. 1, pp. 27–41
doi:10.23743/acs-2018-03

Submitted: 2018-01-06
Revised: 2018-03-05
Accepted: 2018-03-25

blockchain, cloud, datacenter, cluster

Maciej NABOŻNY[*]

ASYNCHRONOUS INFORMATION DISTRIBUTION AND CLUSTER STATE SYNCHRONIZATION

Abstract

This article describes issues related to information distribution and cluster state synchronization in decentralized environments with inconsistent network topology. The main objective of this study was to create a set of rules and functional requirements to build a fault-tolerant framework based on Blockchain for creating applications in decentralized and distributed environments, regardless of the underlying cluster's hardware.

1. INTRODUCTION

Proposed principles could be used as base for many systems, including IT systems, cluster management software, IoT systems and so on. Thus, this paper will discuss only theoretical aspects of distributing information in safe, reliable way.

To define system distributing information across environment, one should define at first basic, necessary components of such system. First, basic element is Node. This should be an individual entity capable of making decisions and interacting with other nodes in the environment. The way these nodes communicate with each other is a Connection. One connection enables one-way communication between two nodes. To provide bidirectional communication between two Nodes, it is required to provide double Connections.

All the above components together are called Clusters. A cluster is a set of connected Nodes and Connections between them, that form a partial or full mesh. This grid can be described using an oriented graph, in which each graph node is able to reach any other node on the cluster. A divided cluster (also described as a split-brain configuration) is a situation in which at least one node is unable to reach at least one other node in the cluster.

[*] cloudover.io ltd, 590 Kingston Road SW20–8DN London, United Kingdom,
+48 511 912 775, maciej.nabozny@cloudover.io

Notification is full information about the change of state of one Node in a Cluster. Notifications are generated by a node that changes its state and broadcasted to all other nodes in the cluster. Nodes can also forward received from neighbors notifications to other nodes, to provide notification routing and thus, redundant communication channels.

By state of the Node we can define the internal state of the resources driven by the Node's logic. Information can be understood as a description of such a state known by local and remote Nodes. Thus, the information about the Node is strongly related to its state, but at some point it may be contradictory.

2. PROBLEM FORMULATION

To create a solution for the described problem of distributing information and synchronizing state of nodes in entire cluster, without a centralized point, it is necessary to build a system that solves the following problems:

- How to handle the "split-brain" of cluster?
- How to handle rejoin of cluster?
- How to authenticate notifications?
- How to limit access to resources?
- How to trust newly attached nodes or objects to cluster?
- How to handle joining two existing clusters?
- How to handle conflicting updates of data after rejoin?
- How to avoid any additional communication channels to use only notifications for communication?
- How to track history of database and all changes over time?

The implementation, which covers all of the above issues, will define a fully functional, event-driven platform for creating distributed systems on the verge of Blockchain, public key infrastructure, and agent technology.

3. SOLUTION

3.1. Information schema

In most distributed systems the database is the central point of application cluster (Fernstrom, Narfelt & Ohlsson, 1992). Proposed approach assumes to keep local copy of database at each node and store information locally. Such organization of information may cause that each node may have different version of information, according to the state of cluster, time and its location among other nodes. Thus, by data synchronization we will define striving for make data sets the same on all nodes. In some cases it will be necessary to accept information inconsistency in different parts of cluster.

To organize the information into logical structures, in programming languages we could define classes and store information as objects (Lippman, 1996). The same idea was proposed here, to group information into objects inside database. Within this article we could redefine an object in context of described system. The object will be a set of named properties related to one logical kind, assigned to the one node. To store one property we should store:

- entity type,
- entity identifier,
- field name,
- field value.

In opposition to classes, proposed solution does not guarantee the schema of information. Once defined entity could have different fields at different nodes, dependently of its version or state. Such approach makes possible to upgrade whole system and change its information organization.

With such approach we could use key-value data store as the underlying database and store information about all objects in database in form:

- Key: entityType:entityIdentifier:fieldName
- Value: the value of field

Such underlying data schema makes possible to synchronize data at field level and manage privileges, on object level. Moreover, with above design one can make inclusion of objects, without any additional requirements of schema. Each included object gets a parent-object's identifier as prefix of this object's keys.

3.2. Split-brain and re-join of cluster

The split brain problem (Davidson, Garcia-Molina & Skeen, 1985) covers situation, when communication between part of cluster or one node and rest of the cluster is temporarily interrupted. Such situation causes the data on particular nodes is not synchronized properly. This leads to situation where node or part of cluster is not up to date with its data, to the rest of cluster. Split brain situations are hard to handle in systems, especially in case when logics running in cluster has to take a decision based on its data.

Described in this chapter split brain is especially dedicated for handling by systems based on blockchain like described here instead of standard architectures and databases, where split brain and rejoin should result the consistent database. Also we won't discuss here the consensus algorithms used by blockchain, which were removed in proposed solution, what will be discussed later.

Most often, split-brain configuration could be caused by non-redundant hardware connecting parts of cluster:

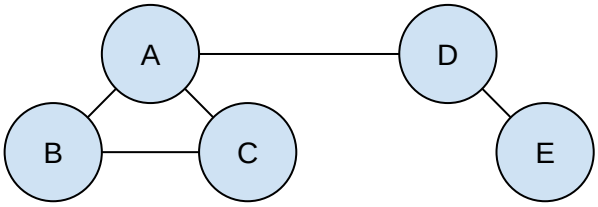

Fig. 1. A cluster with not fully redundant connections

It is possible to indicate four weak points shown in cluster described in Fig. 1. (regardless of the directionality of connections):

- A to B connection,
- D to E connection,
- A notification exchange (routing) from D,E nodes to B,C nodes,
- D notification exchange (routing) from E to A,B,C nodes.

Failure of any point from above list will result split-brain configuration. For example, failure of node A, shown in fig. 2 will split whole cluster to two, sep-arate clusters shown in fig. 3.

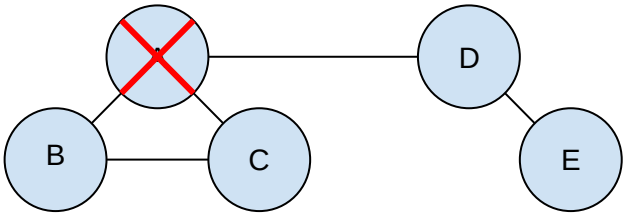

Fig. 2. Failure of node A

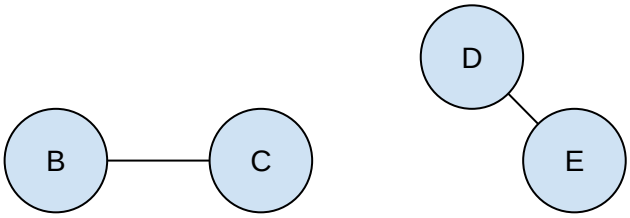

Fig. 3. Cluster topology with outage of node A

Let's assume, that all information related to the cluster state will be available on all nodes till the split-brain, through local databases. In this way, notification exchange between nodes is running till outage of Node A. Moreover, each node after this failure will keep full copy of all information collected till split-brain.

Ongoing lifecycle of cluster could influe to the information stored on particular nodes and generate new notifications related to changes in node's state. For ex-ample, changing the state of node B will create a change in local information.

This will create new notification. In cluster configuration shown in fig. 3, such notification will be shared by node B only to node C. No other possible route for updates is not available here.

Approach to store information locally, not in centralized database for cluster will prevent disconnecting nodes from data source in case of cluster connectivity failure. This could decrease performance of synchronization and propagation at node level, and decrease capacity of overall cluster.. Also duplication of data across all nodes could increase the costs of storing data and hardware in comparison to centralized system.

Handling information by centralized database results that any nodes temporarily detached from cluster make inconsistency quickly. This raises following questions:

- how should the online part of cluster react on changes on re-connected nodes,
- how should re-connected nodes react on changes in online part of cluster,
- how to handle conflicting updates of common parts (i.e. one of online node's changes state of disconnected node during split).

Proposed architecture is oriented to be data driven and to handle any changes of information by local Node's logics. It means, that changes in in information (broadcasted notifications) trigger logics. This is in opposite to: API > logics > database model, which is known from common architectures, where exposed API endpoint triggers execution of logics and then, logics updates central database. Event/Data driven architecture focused on reacting on data updates simplifies handling the rejoin. At the beginning of cluster's lifetime one of nodes should create the first notification about change in database. Each ongoing notification will be marked as successor of the previous one, by marking this fact on one of notification's internal fields. Thus, when all nodes are online, notifications are spreaded almost immediately. Such approach makes the chain of notifications, ordered by internal notification's fields. This idea was taken from the Blockchain (Nakamoto, 2008) database. The difference is that in some cases, chain could become a tree of notifications (or coexisting chains), when two or more nodes are concurrently changing database in the same time. The simple chain of notifications is shown in fig. 4. All notifications are ordered one, after another, by pointer to previous notification.

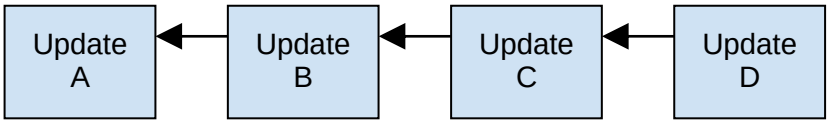

Fig. 4. Chain of notifications

In case of outage of node(s), new updates are not present in its chain. After rejoin, such node will receive only last new update.

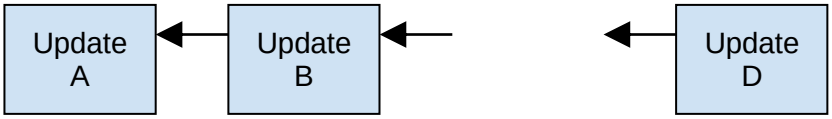

Fig. 5. Incomplete chain of updates, known by node disconnected after receiving update B

Shown in fig. 5 sample chain, represents notifications known by disconnected and then reconnected node. Latest received before outage notification, is notification B. After rejoin, this node will receive ongoing notification D, which points to notification C as its predecessor. Such notification is not known for rejoined node, so it should broadcast request for retransmission of notification C to all its neighbourship. In this way, rejoined node could track all changes made during its outage. Also, in case of new notifications of changes made during outage, this node could retransmit this changes to the rest of cluster. Proposed approach makes greater overhead in comparison to the central database, but makes easy to handle rejoin of any nodes and track historical state of database.

3.3. Handling join of new nodes and joining clusters

Described above way of handling rejoin of particular nodes after split brains applies to the joining new nodes and joining two or more clusters together too. Once new notification appears in node, or new cluster, all previous notifications should be fetched from older part of cluster as the notification's predecessors. There is no additional notification of join for existing cluster, thus all existing nodes could know notifications from new node or newly attached cluster immediately, after first change of information

3.4. Notification broadcasting

To provide better redundancy and failure tolerant cluster of nodes, it is recommended to use physical connections in redundant configuration. Additionally, algorithm responsible to broadcasting new notifications should use all possible connections to broadcast notification. Such approach makes cluster highly available and resistant for failure of particular connections between nodes. This approach could be optimized for better performance, but the it could guarantee lower availability level. Such approach is acceptable, until we need to handle connection failures as the primary functionality.

3.5. Handling conflicting notifications

Due to proposed architecture specification, many nodes could modify simultaneously one information in cluster. Most popular approach suggests to use locks and mutexes (Courtois, Heymans & Parnas, 1971) to avoid concurrency. Proposed solution shifts responsibility of handling such conflicts from database to the logics and notification ordering. Once order of notifications is known, there is no need to deal with concurrent modifications. Each node will process this notification, which is first in chain and modify information respectively to this order.

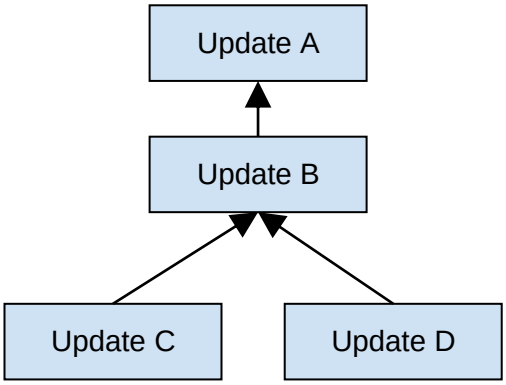

Fig. 6. Two conflicting updates, C and D

In situation of simultaneous notifications, with the same predecessor (fig. 6), logics on each node receives information about conflict. It is certain, that one of two notifications will arrive to node faster, so another will be marked as conflicting. The decision what to do should remain only to the logics at particular node, however simultaneous modification of one resource (i.e. setting value of single variable) is only fraction of such notifications. Most of split brains should produce branches in notification chain, without conflicts. Thus, the information schema in cluster should be designed in way to minimize risk of concurrent modification of the same resources.

The only problematic scenario is when two nodes modify the same resource in database. It is not possible to make decision in advance, so it is necessary to leave this decision to the logics.

3.6. Security of cluster and its data

In order to ensure a fully functional system design, it is necessary to solve the security problems. The proposed system will be an open database, like Blockchain (Nakamoto, 2008), with cryptography-based security. Thus, the permission to read any part of the information will be granted to each node connected to the cluster.

To secure write access to objects, one should use the asymmetric keys to provide signature-based authentication of each notification. Thus, to each object in cluster should be associated with its public and private key. Private key should be known only for the node, which creates object. Public key should be shared with all nodes in cluster with first notification, related to this object creation. This will allow each node in cluster to verify all ongoing notifications related to certain object. By storing list of authorized public keys related to object, we could also grant and revoke permissions to modify this object. Authorized objects changing information related to another object shall sign such notification with own private key, authorized in modified object.

The special case are object's fields which should be available for everyone. Then, the logics should accept notifications without matching signature or to recreate own, signed notification with the same information. Such approach could be used to allocate resources and leave verification to resource handler in cluster. If node's logics could accept such resource allocation, it could re-sign such notification with own, authorized key and re-share with cluster. Otherwise, notification won't be re-signed and it will be ignored by all other participants of cluster.

Once each object has public-private key pair assigned, it is possible to secure the access to object's data. It could be done by encrypting particular fields of object with provided public keys. By using mechanisms known from various encryption systems (i.e. dmcrypt ("DM-Crypt project", 2018) in Linux or GnuGPG ("GNU Privacy Guard project", 2018)), we could grant access to data for multiple nodes in cluster in the same way.

Above two assumptions make access for reading and writing data into the database complete. We could define how to grant access to modifying objects, share data in secure way and handle unsigned notifications.

To prevent spoofing new objects we could make object's ID related to its public key. The relation could be done with one-way hashing function, like SHA (*Secure Hash Signature Standard (SHS)180-2*, 2002) or other. Thus, once object is created with public-private key pair, the ID is the hash of this public key. With this assumption it is impossible to recreate the same object with the same ID and other key pair in reasonable period of time.

3.7. Trusting new objects and nodes

To make possible to authorize all new objects and nodes in cluster, it is necessary to provide some mechanisms against this problem. Thus, mechanisms known from X509 (Cooper, Santesson, Farrel, Boeyen & Housley, 2002), OpenPGP (Callas, Donnerhacke, Finney, Shaw & Thayer, 2007), or other standards could be implemented. Once the object's key is signed by some authority, it could be trusted. There is no other way to automate this process with assumption that in any moment of time, cluster's infrastructure could be splitted.

Otherwise one should have centralized service, verifying the authenticity of objects in database or doing it manually.

3.8. Tracking changes in database

Described assumptions of this system architecture makes tracking history of whole cluster trivial task. Once each node has all notifications, at each point of time we could recreate whole history of cluster's state and information related to it.

3.9. Database as the communication channel

In most IT systems, database acts as the data store. In most cases dedicated interface is responsible for communication with external services. Between interface and database we have layer of logics, which validates requests incoming from interfaces (API) and modifies the database.

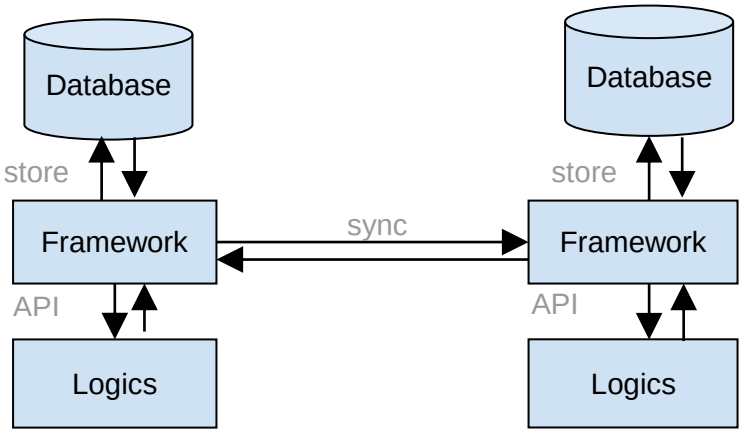

Fig. 7. Architecture of system using notifications as the communication channel

In proposed architecture it is recommend to use fact of information change – the notifications – as the communication channel (fig. 7). The mentioned communication channel in fact is not a database, but the framework intermediating communication with the data store. As mentioned in previous chapters, each node has logics, responsible for validating incoming notifications and making decisions about applying them to local information. From the other side, once state is changed at one node, the notification is broadcasted to other ones. Thus, database updates, together with described authorization mechanisms are an alternative to creating interfaces for communications. For example, changing state of one object in database will create notification of this fact broadcasted to all neighbors. If implementation of this concept provides broadcasting such updates

to all neighbors, and all neighbors will retransmit it to own neighbors, the very strong redundancy will be provided. Moreover, using only proposed system as communication channel for all purposes, creates redundant and fault tolerant channel to communication. However still, it is highly independent on the cluster's physical networking. Handling updates in described in this paper way (chapter 3.A) also defines how to split-brains and re-joins of cluster will be handled.

4. EXISTING TECHNOLOGIES

4.1. IP network and ISO/OSI models

First technology, worth to compare with is the IP networking (in 4 or 6 versions). Basic principle is to provide connectivity between hosts (in our case nodes) and deliver messages in form of packets (notifications). With dynamic routing we could provide redundant paths inside IP networks and make it highly resistant for failures of single points.

However at this point our similarities are end. Of course we could extend IP networking with additional layers, protocols and tools to provide more similar way of communication, but, without implementing application layer of ISO/OSI we cannot do:

- security and authentication – exception is the IPSec protocol, but requires a lot of preconfiguration, what is in opposite to next point,
- autodiscovery and autoconfiguration,
- data store,
- data recreation/database history tracking,
- no default redundancy – configuring redundant connections requires additional routing configuration in whole network.

Thus, for purposes of the discussed architecture, IP network was insufficient solution, however it was used as the backing technology for connectivity. Main difficulty is total lack of data store, combined with cryptographic security layer, which is required in such systems. However some ideas known from IP networks was implemented in proposed architecture.

4.2. Blockchain (including Ethereum, Bitcoin network and other)

Most similar solution to proposed, which has the most common points with described here system is the general Blockchain technology (Nakamoto, 2008). It covers most requirements, mentioned in introduction, but has one, major difference. Blockchain was designed as the base for cryptocurrencies, so its main purpose is to securely make transactions between participants without any centralized point. One of most unique features of blockchain technology is the way

how this database confirms the transactions across the network, known as consensus algorithm. This mechanism is used to create one, final version of database each ten minutes (approx.) by voting. Nodes across network (known as "miners") are used to calculate hashe of block of transactions and thus make cryptographic confirmation of transactions in time. In Bitcoin's network such hash should start with certain amount of zeros at beginning, by calculating SHA of block joined concatenated with random data, resulting proper hash. In Bitcoin this is called proof of work, due to generating hash requires a lot of computational work. In other decentralized systems other mechanisms are used (i.e. proof of stake). This mechanism is most important part of Blockchain and other decentralized systems, what makes it hard to forge.

Ethereum (Eyal, 2017; Tschorsch & Scheuermann, 2016; Wood, 2014) has the similar approach to the verification of smart contracts as the Bitcoin network. The difference is that miners in Ethereum network validate small programs (called smart contracts), instead of validating transactions. Thus, implementing Ethereum as the consistent, asynchronous information distribution system is not possible without large modifications of protocol. Additionally, both technologies (Ethereum and Bitcoin) are based on the same principles of underlying Blockchain. Moreover, the Blockchain's architecture has multiple levels (participants, ledger services, miners). Discussed here clusters should be simple, with flat architecture of nodes with the same privileges.

Above approach makes the Blockchain a great solution for currency purposes. In distributed environment nobody can undo the transaction. Also nobody can forge recent transactions. In context of described system, Blockchain's certainty is a problem. In case of split brains and rejoins we want to deal both parts of cluster, with all changes created by them. Finally, in proposed solution each node decides what part of data and updates is trustworth and which part should be ignored.

4.3. Agent systems

Agent systems, in traditional, scientific approach implement the agents, which are something more than objects. Agents have logics (Zhang & Zhang, 2004) and are able to take decisions based on incoming data, which could be incomplete or incorrect. Most agent systems could take decisions with a certain margin of error.

Thus, described in this paper system is most familiar with the agent systems, but extends it by adding some principles dedicated to drive clusters of nodes and handles various problems not present in agent systems.

5. RESULTS

Described functional requirements and ideas were implemented as stand-alone framework for creating distributed applications in form of library. The library could be found at the project's website ("Dinemic project", 2017) and Github repository ("Dinemic code repositories", 2017). Developed project is in the beta version at the moment of writing this article and is being tested to provide best stability and security, using all described in this paper ideas.

Created framework is designed to hide all complexity of technology, cryptography and networking from developer and provides Event Driven Development approach to creating distributed applications. Except the described functional assumptions, the features of solutions are:

- proxy to the underlying database and full ORM for C++ language – objects created through framework could store data in local data store automatically,
- automatic broadcasting notifications of changes in database over network – all updates on local database are sent to all neighbors over network,
- digital signatures of notifications, data encryption and control of incoming notifications by event handlers – before notifications are received and processed by framework, application can define its own event listeners, to take own actions before or after creating, updating or deleting objects. Event handlers also can prevent unauthorized changes in whole database, objects or particular fields of objects. All behavior could be defined by each application in cluster, what can result in very different database contents on various network nodes.

For applications using dinemic framework there is dedicated application skeleton, which guides developer to use it in proper way. This way allows to easily design applications as event driven. The base class for application – DApp class can be used to create basic models of applications and define necessary models within. Methods of this class (launch and oneshot) are dedicated to handle application installation, short actions (i.e. read or update database contents) and to remove objects when application is removed from node. Models created within the dinemic framework should use inheritance of DModel and DField classes to handle database readings and modifications. Additional class DAction is the interface to creating own handlers of changes in database across whole network. All updates incoming to node can trigger execution of DAction instances defined by user. In this way, application is able to define how to act when database changed fields of certain object at other node. Due to complexity of overall system, usage of al above classes in accordance with documentation is highly recommended. A wider description of framework API remains in documentation.

Implementation of whole framework is done in C++11 language. Beyond the ORM and described event handling mechanisms, the additional two types of configurable drivers were used. First one is the storage driver, which is used to store data in local node's data store. This allows to use on each node different database or key-value store to keep data. Available at this moment storages are: MemoryDriver, which stores data in node's RAM and RedisDriver, which uses Redis server and its key-value store. This driver guarantees more data safety in comparison with MemoryDriver. Second configurable driver is synchronization driver, which should used by each node in the cluster to communicate changes in local database. Synchronization driver monitors all changes in database made by local application and listens for updates generated by other nodes. Its role is also to handle all cryptography aspects - validation of digital signatures, chain verification and notification generation.

Beyond the development of framework itself, strong emphasis were put on aspects of security. During development, a lot of tests were made to confirm framework assumptions and cover all described in this paper functional requirements. Tests are designed to cover all units of framework and also to test its behavior in various network configurations. Additionally, performance tests were made, however publication of them is purposeless, due to the size of cluster, logical design, used hardware and driver set has very high impact on results. Author was not able to find any referral method to compare it with existing system.

6. CONCLUSIONS

Described in this paper assumptions of asynchronous, consistent information distribution for decentralized systems could be treated as the extended idea of agent system combined with the Public Key Infrastructure ideas and cryptography. However mentioned here problems and solutions will allow to create fully functional systems, without any single point of failure known from large scale systems. Described solutions could be applied to the particular applications in Computer Science context, to create the IoT-like systems, or to the whole computing clusters to provide consistent communication between them. The appliance of this solutions depends on needs and shown here assumptions are main guidelines to create decentralized, safe solutions.

In the comparison to the most popular in recent time decentralized system – Blockchain (and similar), this solution abandons one of it principles – the confirmation of transactions by proof of work mechanisms ("Proof of work explanation, Bitcoin project documentation", 2008). In described solution, most important principle was to authenticate origin of changes in whole cluster and trusting them absolutely. Main purpose of that was to trust modifications made by own applications in cluster and prevent unauthorized changes.

Purpose of creating this concept is to provide tool for creating distributed computing clouds, without any centralized point, but still with all basic functionality: storage, networking and virtualization. The main disadvantage of proposed solution and approach to handling management of cluster state is the large overhead of computations on each of cluster nodes. For systems handling up to thousands of virtual machines the amount of changes in central database is not so high to be problematic for described system. However for applications generating large volumes of data and changing data very frequently, this approach might be too slow and complex. In such cases, more appropriate might be using standard architectures, like microservices with strong emphasis on hardware redundancy (Balalaie, Heydarnoori & Jamshidi, 2016; Viennot, Lecuyer, Bell, Geambasu & Nieh, 2015).

ACKNOWLEDGEMENTS

All research and related work were funded and supported by cloudover.io ltd. company. Intellectual property for tools and libraries developed during research remain with cloudover.io ltd. company. The author would like to thank to Roman Krzanowski and Krzysztof Boryczko for theirs help, the Polish Linux Users Group ("Polish Linux Users Group", 2000) for promoting the project and research and the Crypto@Cracow Meetup ("Meetup Group Crypto@Cracow", 2016) group for help.

REFERENCES

Balalaie, A., Heydarnoori, A., & Jamshidi, P. (2016). Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture. In *IEEE Software* (*33*(3), pp. 42–52). USA: IEEE. doi:10.1109/MS.2016.64

Callas, J., Donnerhacke, L., Finney, H., Shaw, D., & Thayer, R. (2007, November). OpenPGP Message Format. Retrieved from https://tools.ietf.org/pdf/rfc4880.pdf

Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., & Polk, W. (2008, May). Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. Retrieved from https://www.rfc-editor.org/rfc/pdfrfc/rfc5280.txt.pdf

Courtois, P. J., Heymans, F., & Parnas, D. L. (1971). Concurrent control with readers and writers. *Communications of the ACM*, *14*(10), 667-668. doi:10.1145/362759.362813

Davidson, S., Garcia-Molina, H., & Skeen, D. (1985). Consistency In A Partitioned Network: A Survey. *ACM Computing Surveys*, *17*(3), 341–370. doi:10.1145/5505.5508

Dinemic code repositories, (n.d.). Retrieved February 1, 2018, from https://github.com/cloudOver/libdinemic

Dinemic project, (n.d.). Retrieved February 1, 2018, from https://dinemic.io

DM-Crypt project, (n.d.). Retrieved February 1, 2018, from http://www.saout.de/misc/dm-crypt

Eyal, I. (2017). Blockchain Technology: Transforming Libertarian Cryptocurrency Dreams to Finance and Banking Realities. In *Computer* (*50*(9), pp. 38–49). USA: IEEE. doi:10.1109/MC.2017.3571042

Federal Information Processing Standards. (2002). *Secure Hash Signature Standard (SHS) (FIPS PUB 180-2)*.

Fernstrom, C., Narfelt, K.-H., & Ohlsson, L. (1992). Software factory principles, architecture, and experiments. In *IEEE Software* (*9*(2), 36–44). USA: IEEE. doi: 10.1109/52.120600

GNU Privacy Guard project, (n.d.). Retrieved February 1, 2018, from https://www.gnupg.org/

Lippman, S. B. (1996). *Inside the C++ Object Model*, 1st edition. USA: Addison-Wesley Professional.

Meetup group Crypto@Cracow. (2016). Retrieved February 1, 2018, from https://www.meetup.com/pl-PL/Crypto-Cracow/

Nakamoto, S. (2008, October). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf

Polish Linux Users Group. (2000). Retrieved February 1, 2018, from https://linux.org.pl

Proof of work explanation, Bitcoin project documentation. (2008). Retrieved February 1, 2018, from https://en.bitcoin.it/wiki/Proof_of_work

Tschorsch, F., & Scheuermann, B. (2016). Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies. In *IEEE Communications Surveys & Tutorials IEEE.* (*18*(3), pp. 2084–2123). USA: IEEE. doi: 10.1109/COMST.2016.2535718

Viennot, N., Lecuyer, M., Bell, J., Geambasu, R., & Nieh, J. (2015). Synapse: a microservices architecture for heterogeneous-database web applications. In *EuroSys'15, Proceedings of the Tenth European Conference on Computer Systems, Article No. 21*. USA, New York: ACM. doi:10.1145/2741948.2741975

Wood, G. (2014). Ethereum: a secure decentralised generalised transaction ledger. Retrieved from http://gavwood.com/Paper.pdf

Zhang, Z., & Zhang, Ch. (2004). Basics of Agents and Multi-agent Systems. In *Agent-Based Hybrid Intelligent Systems. Lecture Notes in Computer Science* (pp. 29–33). Berlin: Springer.

Applied Computer Science, vol. 14, no. 1, pp. 42–52
doi:10.23743/acs-2018-04

Submitted: 2018-02-10
Revised: 2018-03-20
Accepted: 2018-03-23

line balancing problem, manufacturing simulation, Tecnomaix Plant Simulation, manufacturing, production system modeling

Wojciech DANILCZUK[*]

THE USE OF SIMULATION ENVIRONMENT FOR SOLVING THE ASSEMBLY LINE BALANCING PROBLEM

Abstract

The paper describes the application of a simulation environment (Tecnomatix Plant Simulation) for solving a real manufacturing problem. The studied case consisted in rebalancing the production line with a specified number of operators on the line. The first stage of the study involved determination of the production cycle and key performance indicators. The production system was then divided into work cells. After that, proposed design assumptions were verified via a simulation model.

1. INTRODUCTION

One of the main challenges for modern manufacturing companies lies in tailoring their products and manufacturing methods to market needs (Esmaeilian, Behdad & Wang, 2016). Due to dynamically changing demands, companies must be able to quickly adapt their manufacturing systems to create new product types. New solutions for manufacturing systems should not only be developed in a short time, they should also be fully refined. The use of simulation tools enables the verification of intended changes and testing them in a given simulation environment (Longo, 2010). This prevents costly errors that could arise during solution implementation.

[*] Lublin University of Technology, Faculty of Mechanical Engineering, Department of Automation, Nadbystrzycka 36, 20-618 Lublin, tel.: +48 81 538 4267, wojciech.danilczuk@pollub.edu.pl

This paper provides an example of the use of simulation software for the development and verification of a new concept for the organization of an existing production line. This example comes from the author's professional experience. Solutions for similar industrial problems can be found in (Kłosowski & Kozłowski, 2017). The literature of the subject also provides theoretical studies devoted to problems of this type, for example Gola & Wiechetek (2017) or Danilczuk, Gola & Cechowicz (2014).

2. DESCRIPTION OF THE ASSEMBLY LINE BALANCING PROBLEM

The objective of this study was to rebalance an existing assembly line with respect to both layout constraints (its architecture could not be changed) and those of precedence (technological route). The manufacturing plant's management wanted to change the number of workers working on this line. Consequently, the allocation of workstations to workers had to be changed. The entire assembly process is described below. A schematic illustration of the above technological constraints is given in Figure 1.

The investigated assembly line is dedicated to the production of wooden windows. Although some work is done using power tools (e.g. drills), most tasks are performed manually. The first stage of the production process is a parallel assembly of two main parts, A and B. Both parts are assembled in two stages. Subpart A is formed into part A, and between this operation there is a buffer place for one part. Subpart B and part B are assembled in a separate zone. The next step of the technological route consists in fixing parts A and B together and thus creating part C. In this segment of the production line, the workers move parts manually. After that, part C is put on the conveyor. The main material flow takes place in a line segment consisting of the conveyor and an additional assembly station where accessories are attached to the main product. The first operation on the conveyor cell is to attach accessories 1 and 2 to part C. From there part C gets to a quality control station (where it is provided with accessories 3). Another segment of the line is a cleaning station. The last stage is a packaging station. Prior to packing, the worker has to prepare a box and product-protecting elements (e.g. styrofoam corner protectors) as well as to print manuals and documentation.

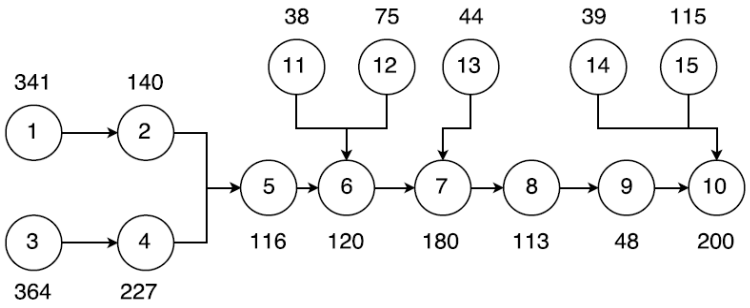

Fig. 1. Technological constraints

The first stage of the study consisted in the revision and validation of the existing technological route. Technological route documentation includes the description of all work tasks that must be performed at every workstation (e.g. drilling holes, gluing corners), the assumed time of every operation and Gantt charts for all operations (Fig. 2). The purpose of validation was to reveal the differences between the operating times specified in the documentation and actual operating times of the line. It was decided that the assembly time of every operation on the line would be measured in order to update the technological documentation. In addition, time measurements were used to calculate new operation cycle times and to assign tasks to individual workers.

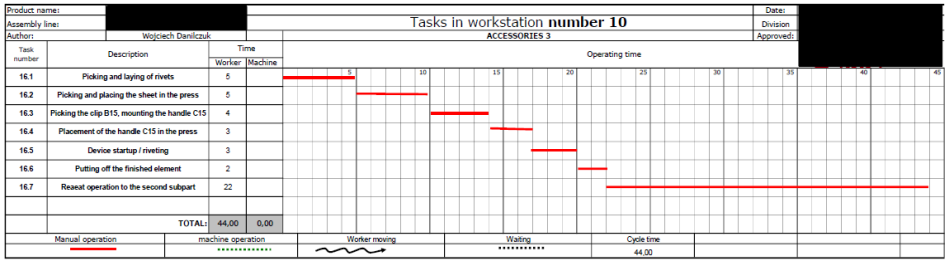

Fig. 2. Example of Gantt chart

3. ASSEMBLY LINE BALANCING

As it was mentioned above, the first step was to update the assembly process documentation, especially with regard to the cycle time of every operation. The measurements were made during normal operation of the line, and the workers were informed about them. They were asked to perform all operations as usual, without rushing or slowing down because of the measuring process. Mean times of all operations (operating time, Top) are listed in Table 1. Once required measurement data was collected, the line balancing procedure was started.

Tab. 1. Operating time

Number	Part / Process	Time (Top)
Operation 1	Subpart A	364
Operation 2	Part A	227
Operation 3	Subpart B	341
Operation 4	Part B	140
Operation 5	Part C	116
Operation 6	Part C + Accessories	120
Operation 7	Accessories 1	75
Operation 8	Quality Control	180
Operation 9	Accessories 2	38
Operation 10	Accessories 3	44
Operation 11	Corner protectors preparation	39
Operation 12	Box preparation	115
Operation 13	Corner protectors assembly	48
Operation 14	Cleaning	113
Operation 15	Packing	200

In single assembly line balancing (SALB), one can distinguish two main types of problem. One is SALB TYPE I, and it occurs when we have a fixed production time and want to find the minimal number of workstations. This problem has been widely described in the literature of the subject (Salveson, 1955; Groover, 2000). There are a few methods for solving SALB TYPE I problems, for example Largest Candidate Rule (Groover, 2000), Kilbridge and Wester Method (Kilbridge & Wester, 1961) or Ranked Positional Weight Method (Helgeson & Birnie, 1961).

The other type of problem occurs when you design a new line for a new product. Nowadays, many lines are modified when a new product is introduced to production. Problems arising from such modifications can be classified as SALB TYPE II – they occur when we have a fixed number of workstations and want to estimate the cycle time (Zemczak, 2013; Grzechca, 2010).

The problem investigated in this study can by classified as single assembly line balancing problem type II. The assembly line in question has a fixed number of workstations, one per every operation. Given the architecture of the line, its layout could not be changed. The technology used in the manufacturing plant enabled the tailoring of the entire line to replicate other line models, depending on the market demand.

The main task was to calculate the cycle time of every operation, assign one worker per every operation (workstation), and thus create a work cell. In addition to this, it was necessary to check whether the line's efficiency met expectations of the manufacturing plant's management; if not, to calculate the value that would fall in line with the expectations. Cycle times were determined with Equation (1) (Zemczak, 2013; Grzechca, 2010)

$$Tc = \frac{\sum Top}{N} = \frac{2160}{6} = 360 \quad (1)$$

where: *Tc* is the estimated cycle time, [s],
Top is the operating time, [s],
N is the number of workers.

One can observe that when the estimated cycle time *Tc* is 360, the operating time of the longest operation (operation 1, assembly of subpart A) is max(*Top*)=364. None of the work tasks can be divided into smaller parts. At this stage of solution design, the author set the cycle time *Tc* equal to 364. This calculation did not take account of transportation time.

The next step was to arrange work cells and to assign workers to work-stations. To do so, the author had to cooperate with the plant's management, as this required taking into account factors such as employee qualifications, the ease of training new employees on particular operations, and staff rotation. Given those limitations, heuristic algorithms could not be applied. The author, in cooperation with the manufacturing plant's management, decided to divide operations into work cells manually, based on an "expert method" and the experience of the managerial and technology staff. Together with the plant's management, the author prepared a worker cell matrix (Table 2). The workload of workers is shown in Figure 3. The work cell diagram (Figure 4) illustrates the allocation of workers to operations.

Tab. 1. Worker cell matrix

	Time (T_{op})	Worker 1	Worker 2	Worker 3	Worker 4	Worker 5	Worker 6
Operation 1	**364**	364					
Operation 2	**227**			227			
Operation 3	**341**		341				
Operation 4	**140**			140			
Operation 5	**116**				116		
Operation 6	**120**				120		
Operation 7	**75**				75		
Operation 8	**180**					180	
Operation 9	**38**				38		
Operation 10	**44**					44	
Operation 11	**39**					39	
Operation 12	**115**						115
Operation 13	**48**						48
Operation 14	**113**					113	
Operation 15	**200**						200
Σ T_{op}	**2160**	**364**	**341**	**367**	**349**	**376**	**363**

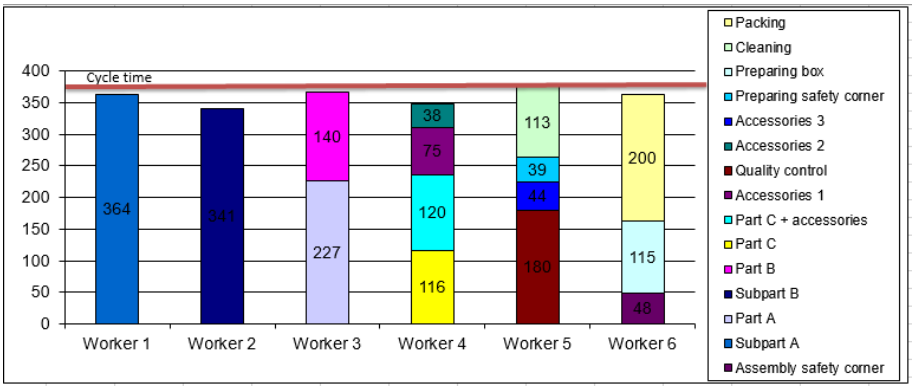

Fig. 3. Workload of workers

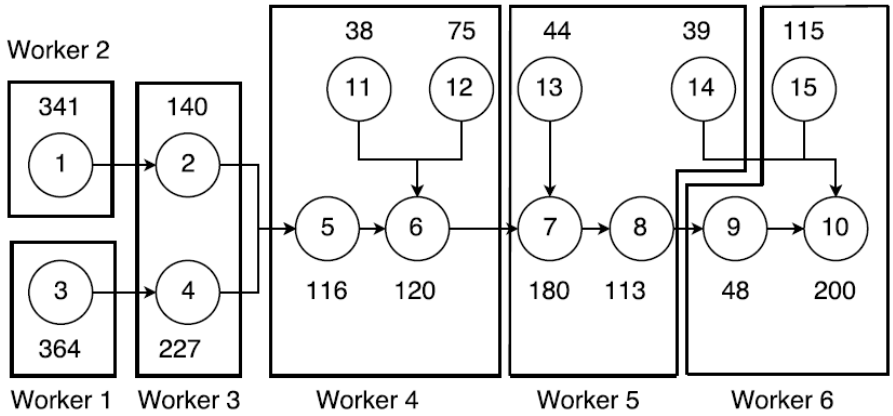

Fig. 4. Work cell diagram

After dividing the operations into work cells, their cycle time and theoretical efficiency were estimated. Based on the work cell matrix, the optimal max(*Tc*) = 376. Because the operations could not be divided into smaller parts, it was not possible to calculate the cycle times of *Tc*=360 nor the time resulting from max(*Top*)=364. The cycle time of the line was assumed to be *Tc* = 376. Based on the assumed cycle time, efficiency indexes (2) were calculated (Scholl, 1999; Grzechca, 2010). An additional variable used in the calculations was a coefficient describing the number of parts manufactured by one employee per shift, KPI (3).

$$EI = \frac{\sum Top}{N * Tc} = \frac{2600}{6 * 376} = 95{,}7\% \qquad (2)$$

where: *EI* is the efficiency index,
Top is the operating time, [s],
Tc is the cycle time, [s],
N is the number of work cells.

$$PpS = \frac{St}{Tc} = \frac{7{,}5 * 3600}{376} \approx 71 \tag{3}$$

$$KPI = \frac{PpS}{N} = \frac{71}{6} = 11{,}83 \tag{4}$$

where: *PpS* is the number of parts manufactured per shift,
St is the shift time, [s],
Tc is the cycle time, [s],
KPI is the number of products manufactured by one employee per shift,
N is the number of work cells.

4. SIMULATION ENVIRONMENT

Following the design phase but prior to the implementation of a new assembly line organization, a simulation was performed in the Plant Simulation software. Since the line worked in accordance with old procedures during the rebalancing process, it was reasonable to carry out the simulation before implementing desired changes. This approach enables the verification of proposed solutions without disturbing the line's operation (Banks, Carson, Nelson & Nicol, 2010). In the event of an error, it is possible to make amendments without interrupting operation of the line and decreasing its performance. What is more, theoretical calculations take no account of manual transportation of parts between the work-stations nor the movement of the workers on the line. In contrast, a simulation model allows for taking these phenomena into account.

The environment used to prepare a model and perform simulation was Tecnomatix Plant Simulation 11. This environment is part of the Siemens software for PLM and digital manufacturing. Plant Simulation enables the simulation, visualization and analysis of manufacturing processes (Bangsow, 2010). The advantage of the software is that it can be integrated with other SIEMENS tools including Teamcenter, Process Simulate and Solid Edge ("Plant Simulation", 2018).

The program enables the introduction of random variables to the simulation, e.g. the supply of raw materials according to preset statistical distributions and the parameterization of individual objects, e.g. the definition of MTTF machines. Thanks to the integration with CAD tools, it is possible to import ready-made machine models and to prepare a simulation model while maintaining

the geometry of objects on the production line. The functionality of standard components can be extended with the tool for writing scripts in the embedded programing language SimTalk (Bangsow, 2010). An example of such a script is given in Figure 5. This enables the preparation of algorithms that control both line operation and events occurring during the process. The program also generates reports on line performance and production line statistics.

A simulation model of the analyzed line is shown in Fig. 6.

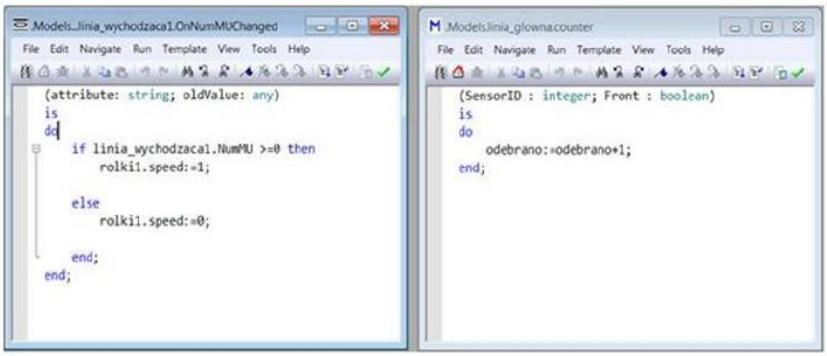

Fig. 5. Example of a script in SimTalk (Danilczuk, Gola & Cechowicz, 2014)

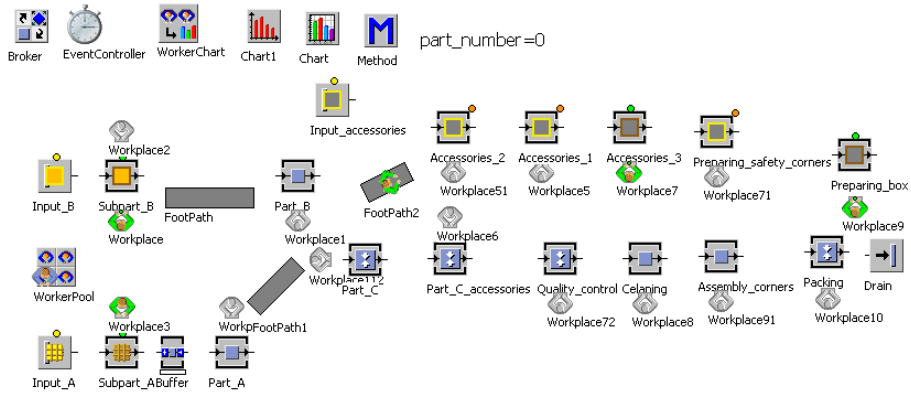

Fig. 6. Simulation model of the analyzed line

5. RESULTS

The use of simulations allowed for testing the proposed solution before implementing it into the production process. In the first stage of the design process, the transportation time between individual operations was not taken into account. In the simulation model, however, this variable was considered.

Additionally, the simulation allowed for examining the line start-up effect. The efficiency of the workers operating the line and the utilization of individual workstations are plotted in Figure 7.

The worker efficiency on all work cells is similar and exceeds 90% of their working time. This indicates a proper balance of the production line. None of the workers is overloaded with assigned tasks.

The use of the buffer between operations 1 and 2 (assembly of subpart A and part A) amounts to 72%, therefore it is justified to maintain it.

The number of parts manufactured per one shift is 68 (assuming that there are no intermediates on the line after start of work). The difference between the number of parts manufactured per shift, *PpS*, obtained in the theoretical calculation (71 items) and that obtained from the simulation (68 items) results from the simplifications made in the calculations. Also, the number of parts manufactured by one employee per shift, *KPI* (3) is different for the theoretical calculation value and that obtained from the simulation – it is 11.83 and 11.33, respectively.

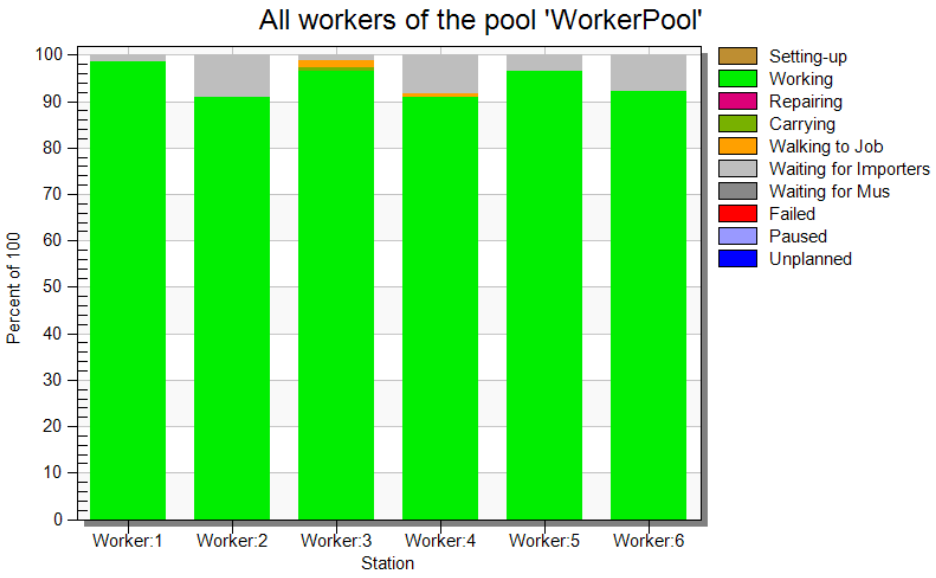

Fig. 7. Worker efficiency

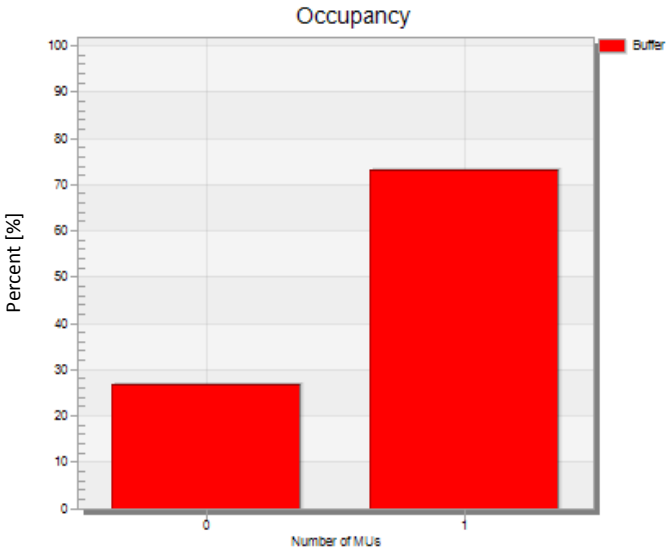

Fig. 8. Buffer occupancy

6. CONCLUSIONS

Although the problem of assembly line balancing is widely described in the academic and industry literature, it still poses a great challenge. This is due to the fact that, under real industrial conditions, even such a basic task as the determination of a production cycle time may occur to be complex.

Simplifications made in the determination of line performance and other indicators may lead to incorrect assessment of the situation and cause disagreements between process engineers, plant managers and production staff. Thanks to the use of simulation tools, the analyzed phenomena could be simulated and visualized, and the target efficiency of the production line was estimated.

Another important aspect of the use of simulation environment is that it enables the verification of design assumptions and the division of the line into work cells before implementing proposed solutions. The possibility of analyzing the proposed solutions in a virtual environment allowed for the verification of their correctness without interrupting operation of the line, which would lead to incurring losses. In addition to this, the use of the simulation software turned out to be an important managerial tool for convincing the management and production staff of new solutions.

REFERENCES

Banks, J., Carson, J. S., Nelson, B. L., & Nicol, D. M. (2010). *Discrete-Event System Simulation*. USA: Prentice Hall.

Bangsow, S. (2010). *Manufacturing Simulation with Plant Simulation and SimTalk*. Berlin: Springer.

Danilczuk, W., Gola, A., & Cechowicz, R. (2014). Analiza konfiguracji linii produkcyjnych na postawie modeli symulacyjnych. In K. Bzdyra (Ed.), *Informatyczne Systemy Zarządzania* (tom 5, pp.25–42). Koszalin: Wyd. Politechniki Koszalińskiej.

Esmaeilian, B., Behdad, S., & Wang, B. (2016). The evolution and future of manufacturing: A review. *Journal of Manufacturing Systems*, *39*, 79–100. doi:10.13140/RG.2.1.2720.0402

Gola, A., & Wiechetek, Ł. (2017). Modelling and simulation of production flow in job-shop production system with enterprise dynamics software. *Applied Computer Science*, *13*(4), 87–97. doi: 10.23743/acs-2017-32

Groover, M. (2000). *Automation, Production Systems, and Computer-Integrated Manufacturing (2nd Edition)*. New York: Prentice hall.

Grzechca, W. (2010). Dualność problem balansowania linii montażowej. In R. Knosala (Ed.), *Komputerowo Zintegrowane Zarządzanie* (pp. 532–539). Opole: Oficyna Wydaw. Polskiego Towarzystwa Zarządzania Produkcją.

Helgeson, W. B, & Birnie, D. P. (1961). Assembly line balancing problem using ranked positional weights technique. *Journal of Industrial Engineering*, *12*(6), 394–398.

Kilbridge, M., & Wester, L. (1961). A Heuristic Method of Assembly Line Balancing. *Journal of Industrial Engineering*, 19(6), 292–298.

Kłosowski, G., & Kozłowski, E. (2017). Optymalizacja wielokryterialna w procesie produkcji mebli. *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, *4*, 101–106.

Longo, F. (2010). Emergency simulation: state of the art and future research guidelines. *SCS M&S Magazine*, *1*(4), 2010-04.

Salveson, M. E. (1955). The Assembly Line Balancing Problem. *The Journal of Industrial Engineering*, *6*(3), 18–25.

Scholl, A. (1999). *Balancing and Sequencing of Assembly Lines*. Physica-Verlag Heidelberg.

Zemczak, M. (2013). Zagadnienie balansowania linii montażowej i szeregowania zadań w systemach produkcji mixed-model. In K. Bzdyra (Ed.), *Informatyczne Systemy Zarządzania* (tom 4, pp. 115–128). Koszalin: Wyd. Politechniki Koszalińskiej.

Plant Simulation. (n.d.). Retrieved 2018, from Siemens PLM Software website https://www.plm.automation.siemens.com/en/products/tecnomatix/manufacturing-simulation/material-flow/plant-simulation.shtml

Applied Computer Science, vol. 14, no. 1, pp. 53–63
doi:10.23743/acs-2018-05

Submitted: 2018-02-24
Revised: 2018-03-05
Accepted: 2018-03-19

fluctuation, prices, machine learning, predictive model, cassava derivative

Odunayo OLANLOYE[*]*, Esther ODUNTAN*[**]

MACHINE LEARNING PREDICTIVE MODELING OF THE PRICE OF CASSAVA DERIVATIVE (GARRI) IN THE SOUTH WEST OF NIGERIA

Abstract

Fluctuation in prices of Agricultural products is inevitable in developing countries faced with economic depression and this, has brought a lot of inadequacies in the preparation of Government financial budget. Consumers and producers are poorly affected because they cannot take appropriate decision at the right time. In this study, Machine Learning(ML) predictive modeling is being implemented using the MATLAB Toolbox to predict the price of cassava derivatives (garri) in the South Western part of Nigeria. The model predicted that by the year 2020, all things being equal, the price of (1kg) of garri will be ₦500. This will boost the Agricultural sector and the economy of the nation.

1. INTRODUCTION

Artificial Intelligent (AI) is an aspect or area of computer science that is widely applicable in solving some fundamental real life problems. It has become a very useful tool in solving real life problems in different areas of life. For instance, AI has been found to be very useful in the field of science, engineering, medicine, bio-technology, cybernetics, mathematics, game theory etc. Its application has become so wide or broad to the extent that it is becoming increasingly difficult to give it a precise or single definition.

[*] Department of Computer Science, Emmanuel Alayande College of Education, Oyo, olanloyeo@yahoo.com
[**] Department of Computer Science, Federal Polytechnic, Ilaro, estherbest2003@yahoo.com

AI could be defined as a science of automating intelligence behavior (Vandal, 2010) .This definition suffers the fact that the word intelligence cannot be absolutely defined. Olanloye (2017) defined intelligence as the capacity of a system to achieve a goal or desired behavior under condition of uncertainty.

Intelligence can also be defined as ability to think, learn, and act appropriately when it is necessary. Just like the way man can think, learn and act appropriately under certain condition because of its natural intelligence, so it is possible to create intelligence (artificial) in machine such that the machine too can solve real life problems in a humanlike manner. Therefore, AI can be defined as an aspect of computer science which makes attempts to create intelligence in machine such that the machine too can solve real life problem in a humanlike manner.

Machine learning is an important branch of AI which is defined as optimizing a performance criterion using example data and post experience (Ethem, 2010). In ML, data plays an important and indispensable role and the learning algorithm therefore acquire its knowledge or experience from data. Hence, the nature of data and the quality of the data will affect the learning and the prediction performance(Wei-lun,2011).Machine learning is used extensively in the area of forecasting and predictive modeling. Many researchers have made series of attempts to use predictive modeling to solve different types of real life problems.

In this research work, an attempt is being made to predict the price of cassava derivative (garri) using machine learning predictive model. The prices were predicted for every 5 years between 1980 and 2050 using periodic and exponential coefficient. Machine learning in MATLAB toolbox was used for the purpose of implementation.

2. STATEMENT OF PROBLEM

Nigeria grows more cassava than any other country in the world and about 70% of cassava produced in Nigeria is processed into garri which has become the most commonly traded and consumed product of cassava (Orewa & Egware, 2012). The trading and consumption of garri is not even restricted to local level as it is being exported to different parts of the world.

Producers and processors are constantly faced with problem of seasonal variation in the product price (Olagunju , Babatunde & Salimonu, 2012) as stated in (Izekor, Alufohai & Eronmwon, 2016). This is one of the major problems of agricultural product – i.e. the price of the product produced by farmers suffers several price variations.

The price of garri continues to fluctuate in the market. This has been a serious concern to both the consumer and the producers of this product. During the season of cassava, the buyer buys garri at lower prices and this is disadvantageous to the farmers. When cassava season is off, the garri buyers buy at higher prices but the producer (the farmers) can only supply in low quantity. Hence, one can

conclude that the driving force that controls the demand and supply of cassava in Nigeria is the price garri. How can we then predict or forecast the price of garri such that the farmers, producers, buyers and consumers can have good idea of the price of the product in the market? This will assist them in the area of planning and decision making. Such prediction will also assist the Government in mapping out good economic policies.

A lot of models (most especially Statistical model) have been developed to solve similar problems with respect to some other agriculture product but no attempt has been made to predict the price of garri using AI approach.

3. OBJECTIVES OF THE STUDY

The main objective of this research is to use machine learning predictive modeling to predict the price of cassava derivatives (garri) in the south Western part of Nigeria.

4. LITERATURE REVIEW

Izekor, Alufohai and Eronmwon (2016) presented a research work on analysis of market integration and price variation in garri market in Edo State. Price behavior between rural and urban market for Garri was examined to determine the price transmission. The method used was statistical and no predictive model algorithm was developed.

With statistical approach, Orewa and Egware (2012) were able to compare the price of Garri in both rural and urban market of Edo State between the periods of 1990 to 2005. No model or algorithm was developed and the research work did not make any prediction.

Iwayan et al. (2010) also predict the quality of cocoa using image processing and Artificial Neural Network(ANN). Though this research work made use of AI principles but it has nothing to do with neither cassava nor its derivatives.

Again, Gurudeo and Tereq (2016) presented a research work on oil price forecasting using univariable time series models. The method used is more of statistics than AI. Though, the developed model was able to make predictions on the oil price. The area of application is in oil sector and has nothing to do with cassava and its derivatives.

Literature also reveals that Faisal and Wumi (2005) made use of AI principle, precisely ANN to predict the crude oilprice trend. The prediction was not on the price of cassava and its derivatives.

Sunday et al. (2014) were able to carry out monthly price analysis of cassava derivatives in rural and urban market in Akwa Ibom State in southern Nigeria. No algorithm or model was developed and there was no prediction. The research only carried out the price analysis using statistical approach and not AI approach.

Ernest (2002) used AI (i.e. Neural Network and Genetic Algorithm) to predict the future price of commodity. The work is not particular about price of cassava or its derivatives, and again NN and GA approach were used. The method used was quite different from the proposed method.

Obe and Shangodoyin (2010) were able to use ANN based model to forecast sugar cane production. This study applied heuristic technique to develop an ANN model to forecast sugar cane production in Nigeria. The model was proved by the authors to have carried out a very accurate prediction. Though, AI principles were used as the method of prediction but the research has nothing to do with cassava and its derivatives.

Going through the literature, one can conclude that prediction of the price of cassava and its derivatives has not been seriously addressed using AI principles. These researchers therefore made an attempt to predict the price of cassava and its derivatives using ML approach.

5. THE CONCEPT OF MACHINE LEARNING

Machine learning can be defined as an automated learning. Learning is the process involved in converting experience into expertise. In Machine learning, the machine is trained with large amount of data to acquire enough knowledge or experience and such knowledge or experience is applied in solving future problems. The input is presented to the learning algorithm in form of training data which represent experience and the output is in form of expertise which usually takes the form of another computer program that can perform some task. It is needed where there is a large amount of data to be considered and when there is need for adaptability.

According to Harrington (2012), the steps involved in developing a machine learning application are:

- **Collect the data:** this could be done personally or though some devices. It could be from RSS or an API. It could be through satellite, remote sensors, internet etc.
- **Prepare the input data:** one should ensure that the data is in the right format. In most cases, the algorithm specific formatting will be taken into consideration. Some algorithm can deal with target variables expressed in form of strings whereas some other algorithms make such variables in form of integers.
- **Analyze the data:** The data to be used should be carefully analyzed. One may need to study the pattern and make some grouping. Analysis again may call for plotting the data in form of two or three dimensional structure. When we have multiple dimensions, technical reduction may be used to reduce the dimension so as to make data visualization very easy.

- **Train the Algorithm:** This is the core area of machine learning where sufficient data should be supplied to the system for training and the system is expected to learn from such data. The type of learning could be supervised or unsupervised type.It is supervised learning if the type of result (output) expected is provided for the algorithm and hence the algorithm works towards obtaining such output. It is an unsupervised learning if there is no target output but the algorithm struggle on its own to recognize the pattern of the input data to carry out some classification to generate the required output.
- **Test the Algorithm:** Having trained the algorithm, it is imperative to test for the level of performance of the algorithm. This will help to evaluate the training process. This is just a way of evaluating the algorithm. If the performance is not satisfactory, one may go back to effect some changes and test the algorithm again until the performance is accessed to be quite satisfactory.
- **Update**: For the purpose of implementation, a notable programming language like python, MatLab, R and so on,are used.

There are different types of machine learning algorithm depending on the type of the problem to be solved. They includes

- **Regression:** This is basically for prediction e.g.predicting the price of a building based on the facilities that are available in the building, predicting the weather condition based on data of the previous weather conditions etc.
- **Binary Classification:** It is also meant for prediction but unlike the regression, the output of prediction is either a yes or no e.g. predicting whether a woman has breast cancer ornot. It groups the women into 2 parts i.e. those with breast cancer and those without.
- **Multiclass Classification:** In multiclass classification, there are many classes, what it does is to predict the most appropriate class for a particular sample. For instance, the algorithm could predict whether a particular student should belong to science, Arts, Commercial or Technical Class.
- **Ranking:** There are set of objects which are to be arranged in a particular order based on predictive model. Examination scores of the students could be arranged according to the magnitude or the size of the scores.
- **Prediction:** Prediction is the act of developing hypothesis or algorithm to forecast future events based on the set of historical events. It is the act of developing a model for the purpose of forecasting future events. A perfect prediction provides insight into the implication of an action and it serves as a metric to judge one's ability to influence or judge future events (Hetemäki & Mikkola,2005; Peralta et al., 2010).

6. METHODOLOGY

Large historical market price of cassava derivative (garri) was collected for a period dating from 1970 till present date from South Western part of Nigeria. The collected data was then compared with the simulated model to determine the actual values of the exponential and periodic factor. A computational machine learning algorithm (regression) was developed with equation 1, 2, 3:

$$Y = sin(cp * t) \quad (1)$$
$$Z = exp(ce * t) \quad (2)$$
$$Zy = Z * (Y + 1) \quad (3)$$

where: cp – periodic coefficient,
ce – exponential coefficient,
t – time variable which is the periodic components of the Garri price,
z – exponential component of the garri price.

We simulate with the various exponential factor and periodic factor. The exponential factor was varied from –0.01 to 0.13 .The periodic factor was varied from 0.8 to 2.2. Comparing with actual data shows that the exponential factor for actual data is around 0.1 and the periodic factor is around 2.0. The fluctuation period is about 5 years and the growth is an exponential function with an exponential growth of 0.1.

The algorithm is implemented using MatLab ML toolbox. Series of efforts were made to obtain the best model which learn from the previous data (change in price of garri over certain period of time) and then predict the future prices for every 5 years.

7. RESULT AND DISCUSSIONS

Series of results were generated and presented in three dimensional visualization showing both the periodical and exponential factors. Some of these were shown in Figure 1 and 2. Results obtained in Figure 2 appears to be better than others. The result obtained shows the price of garri continues to increase overtime. The price of a 1kg of garri was ₦0.05 in 1970. Since then, it has been fluctuating every year. Considering the fluctuation period of 5 years, in 1975, the price was around ₦0.16. In the year 1980, 1985, 1990, 1995, 2000 the prices were 0.5, 1, 2, 7 and 20 naira respectively. Following the same trend, the model predicts that all things being equal, the price of 1kg of garri will be ₦500 in the year 2020. Figure 2, 3, 4 and Table 1 explain further.

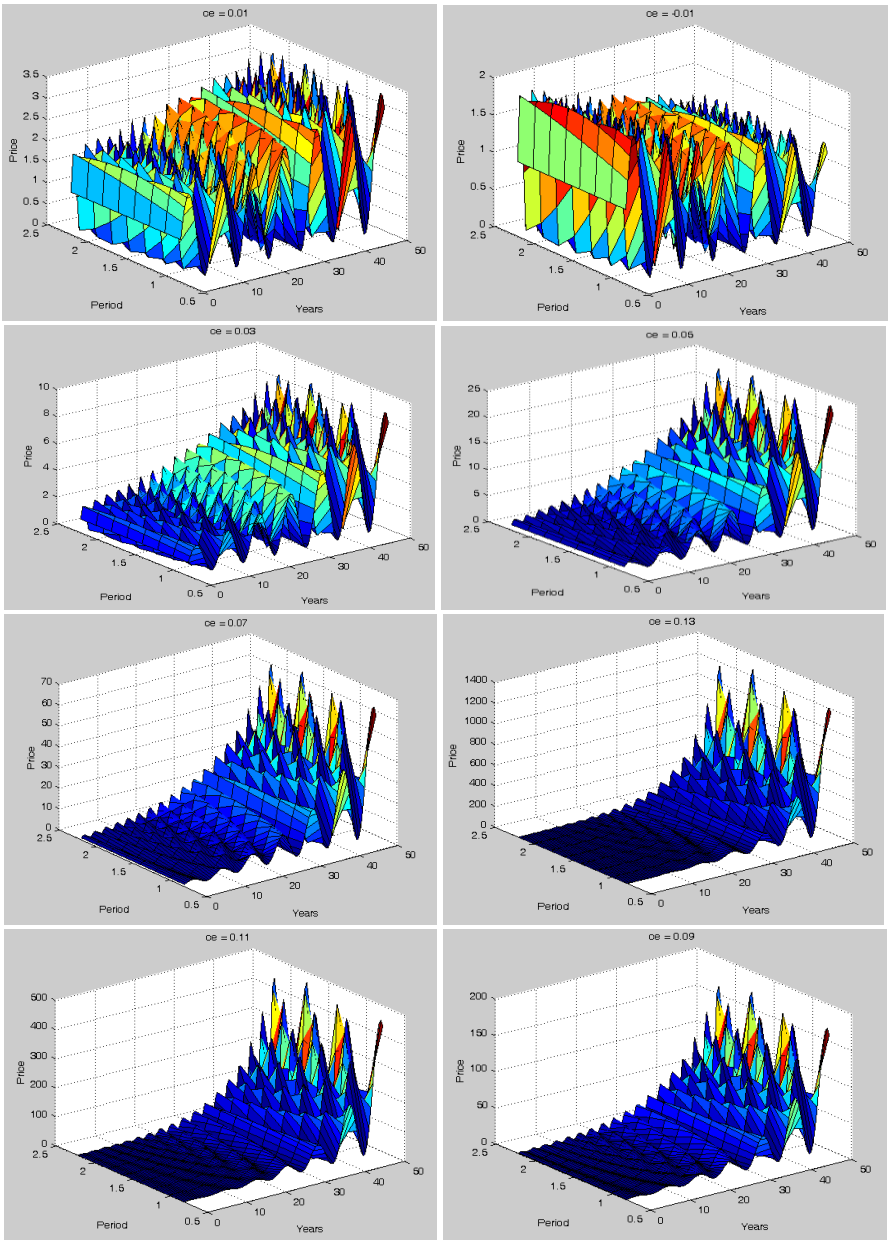

Fig. 1. Sample Models to Predict the Price of Garri

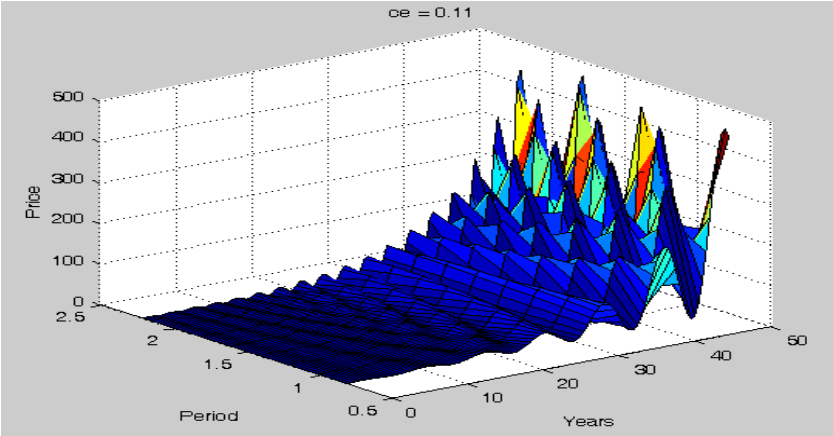

Fig. 2. The most Accurate Model for Garri Price Prediction

The values (prices) of 1 kg of garri in the market and from the model (as displayed in the in the 3 dimensional shapes in Figure 2) were displayed in the Table 1 below. Figure 3 shows the graphical variation in the price of garri as determined by the new model. It further explains the basic characteristics of Nigeria economy where the prices of goods and services continue to increase. An attempt was also made to show clearly the differences between the market price and the model price using simple bar chart as shown in Figure 4.

Finally our model predicts that by the year 2020 the price of 1 kg of garri will be as high as ₦500, all things being equal.

Tab. 1. Comparison of the Model Price with the actual Market Price.

S/N	Year	Model Price (MP) in Naira	Market Price (KT) in Naira
1	1970	0.05	0.06
2	1975	0.16	0.15
3	1980	0.50	0.70
4	1985	1.00	1.10
5	1990	2.00	1.80
6	1995	7.00	6.80
7	2000	20.00	22.00
8	2005	50.00	55.00
9	2010	100.00	95.00
10	2015	200.00	250.00
11	2020	500.00	–

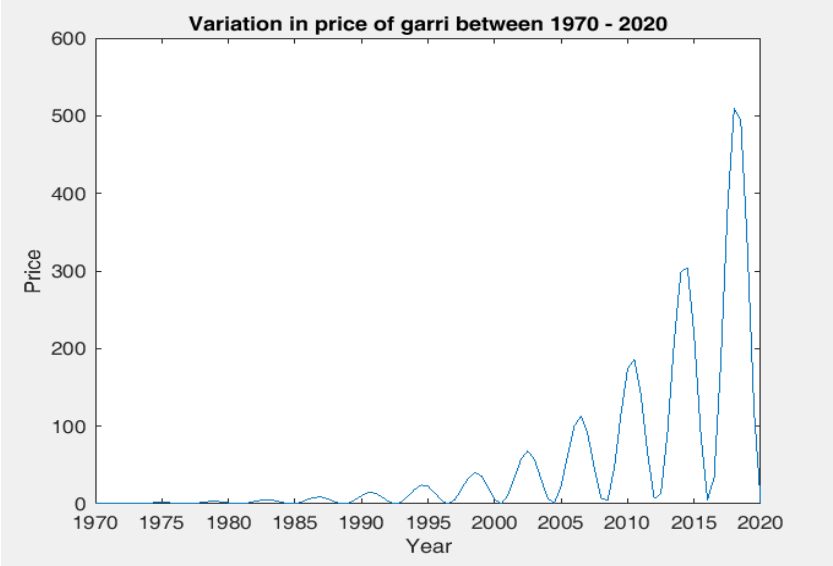

Fig. 3. Graphical Variation in the Price of Garri as Determined by the New Model

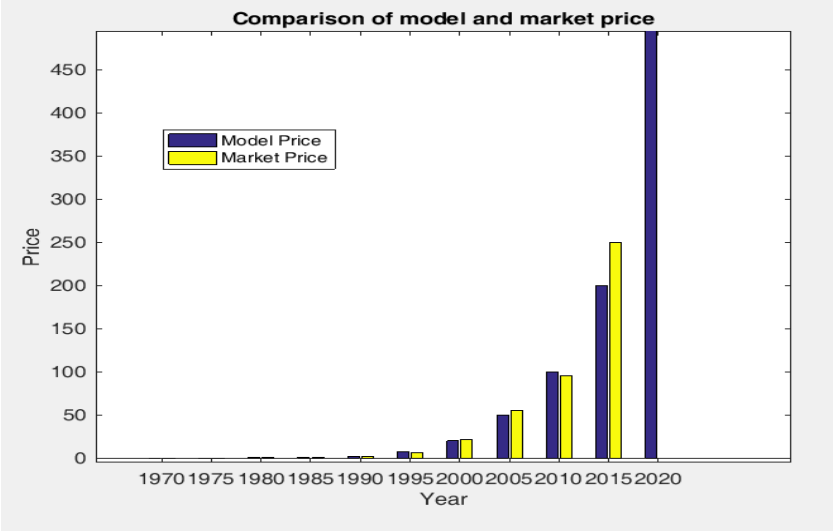

Fig. 4. Comparism of Model and Market Price

8. CONCLUSION

This research work introduced an AI principle precisely ML algorithm to develop a model that can be used to predict the price of garri. The exponential coefficient and the periodic coefficient were used to formulate the model using machine learning algorithm. The implementation was done using MatLab tools. The model was made to predict the price of garri for every five years, starting from 1970 to 2020. It was eventually predicted by our model that by the year 2020, all things being equal, the price of 1Kg of garri will be ₦500. Therefore this research work will serve as a good predictive tool that will be of great assistance to the all the stakeholders in garri industry in term of taken appropriate decisions at the right time. It will boost the nation agricultural sector. It will also assist the government to prepare good financial budget for the nation.

In fact, there will be overall improvement in the standard of living and hence the economy of the nation.

REFERENCES

Ernest, A. F. (2002). *Commodity future price prediction; An Artificial Intelligence approach.* A Thesis Submitted to the Graduate Faculty of the University of Georgia for the Award of Maters of Science.

Ethem, A. (2010). *Introduction to Machine Learning, second edition*. London, England: The MIT Press, Cambridge, Massachusetts.

Faisal, A., & Wumi, I. (2005). Predicting crude oil price trends using ANN modelling approach. In *25th North America Conference*. Denver and Co.

Gurudeo, T. A., & Tereq, S. (2016). Oil price forecasting based on various univariate time series models. *American Journal of Operation Research, 6*, 226–235.

Harrington, P. (2012). *Machine Leaning in action*. Shelter Island, NY: Manning Publication Co.

Hetemäki, L., & Mikkola, J. (2005). Forecasting Germany's printing and writing paper imports. *Forest Science*, *51*, 483–497.

Iwayan, A., Muhamad, S., Andril, K., & Yunindri, W. (2010). Determination of cocoa bean quality with image processing and Artificial Neural Network. *Computer Based Data Acquisition and Control in Agriculture*, *2*(1), 13–16.

Izekor, O. B., Alufohai, G. O., & Eronmwon, I. (2016). Analysis of market integration and price variation in Garri marketing in Edo State, Nigeria. *Nigerian Journal of Agriculture, Food and Environment, 12*(4), 123–130.

Obe, O. O., & Shangodoyin D. K. (2010). ANN based model for forecasting sugarcane production. *Science Publication, Journal of Computer Science*, *6*(4), 439–445.

Olagunju, F. I., Babatunde, R. O., & Salimonu, K. K. (2012). Market structure, conduct and performance of *Garri* processing industry in South Western, Nigeria. *European Journal of Business and Management*, *4*(2), 99–112.

Olanloye, D. O. (2017). *Development of Artificial Intelligence Goeinformatics System for solid mineral prospecting* (doctoral dissertation). Dept. of Computer Science, Nnamdi Azikiwe University, Awka, Nigeria.

Orewa, S. A., & Egware, R. A. (2012). Vomparative analysis of rural and urban market prices for Garri in Edo State, Nigeria: Implications for Food Security. *Journal of Development and Agricultural Economics*, *4*(9), 252–257.

Peralta, J., Li, X. D., Gutierrez, G., & Sanchis, A. (2010). Time series forecasting by evolving Artificial Neural Networks using Genetic Algorithms and differential evolution. In *The* 2010 *International Joint Conference on Neural* Networks (pp. 1–8). Barcelona: IEEE. doi: 10.1109/IJCNN.2010.5596901

Smith, V. (2010). Application of Neural Network in weather prediction*."The Pacesetter"*, *10*(3), 25–35.

Sunday, B., Ini-mfon, V., Samuel, J., & Udoro, J. U. (2014). Monthly price analysis of Cassava Derivatives in Rural and Urban Market in Akwa Ibom State, Southern Nigeria. *Agricultural Science Value*, *2*(1), 48–68.

Wei-lun, C. (2011). *Machine Learning tutorial*. Disp Lab, Graduate Institute of Communication Engineering, National Taiwan University.

Applied Computer Science, vol. 14, no. 1, pp. 64–72
doi:10.23743/acs-2018-06

Submitted: 2018-02-20
Revised: 2018-03-05
Accepted: 2018-03-19

superconducting windings, HTS tape, LabVIEW, superconducting fault current limiter

Rafał KWOKA[*]*, Janusz KOZAK*[**]*, Michał MAJKA*[*]

TESTS OF HTS 2G SUPERCONDUCTING TAPES USING THE LABVIEW ENVIRONMENT

Abstract

HTS 2G superconducting tapes without stabilizer are used for the construction of superconducting fault current limiters. The characteristics R =f(T) of the superconducting tape is essential for design of superconducting current limiters and analysis of their numerical models. The article presents the experimentally determined characteristics R=f(T) of the second generation superconducting tapes SF4050 and SF12050 using the LabVIEW integrated environment. The article describes the method of sample preparation and methodology of testing of HTS 2G superconducting tapes. The LabVIEW integrated environment was used for the testing, thanks to which it was possible to download, process and save measurement data.

1. INTRODUCTION

The special properties of superconductors enable the construction of electrical devices with parameters that cannot be obtained when using conventional materials. HTS 2G tapes based on yttrium have appropriate parameters for the construction of high current and high voltage limiters. In superconducting fault current limiters, the phenomenon of superconducting material retreating from the superconductive state to enter the resistive state is used as a result of exceeding the critical current value *I*c (Kozak, Majka & Kwoka, 2017; Czerwiński, Jaroszyński, Majka, Kozak, & Charmas, 2016).

[*] Lublin University of Technology, Institute of Electrical Engineering and Electrotechnologies, Nadbystrzycka 38A, 20-618 Lublin, rafal.kwoka@pollub.edu.pl, m.majka@pollub.pl
[**] Electrotechnical Institute, Pożaryskiego 28, 04-703 Warsaw, j.kozak@iel.waw.pl

Carrying out tests of HTS 2G superconducting tapes requires the use of appropriate measuring equipment and software. Often, the tests require a novel and flexible tool that enables efficient measurements, data processing, visualization and archiving of measurements and calculation results. The LabView programming environment from National Instruments seems ideal for such applications.

2. LabVIEW ENVIRONMENT

The LabVIEW programming environment (Virtual Instrument Engineering Workbench) is used in tests in industry, research and development, and wherever measurements and data analysis are performed. LabView is a graphical programming language in which icons instead of lines of code are used to create applications. Unlike textual programming languages, where instructions define the way in which a program is executed, LabView uses graphical programming of data flow between nodes and mathematical operations on data ("LabVIEW National Instruments", 2018).

LabVIEW allows one to design a graphical user interface and a source code integrated with it, also created in the graphical programming language G. The program in G language differs fundamentally from programs written in other, conventional (textual) languages, first of all, in the fact that it does not include variables in the explicit form, and the program is structured by the data flow. It also forces the sequence of actions: the specified function (component) of the program will be executed only when all required data will be delivered to it.

The LabView applications are called "virtual instruments" due to their similarity to physical measuring devices, such as oscilloscopes or meters. They consist of two basic components: the front panel and the block diagram, an example of which is shown in Figure 1. The front panel is the user interface. Here, the programmer can place input elements – used to enter data, application operation parameters, such as numerical inputs, knobs, sliders, buttons, and output elements – used to present the results of the application operation, such as numerical outputs, indicators modeled on analog meters ("LabVIEW National Instruments", 2018).

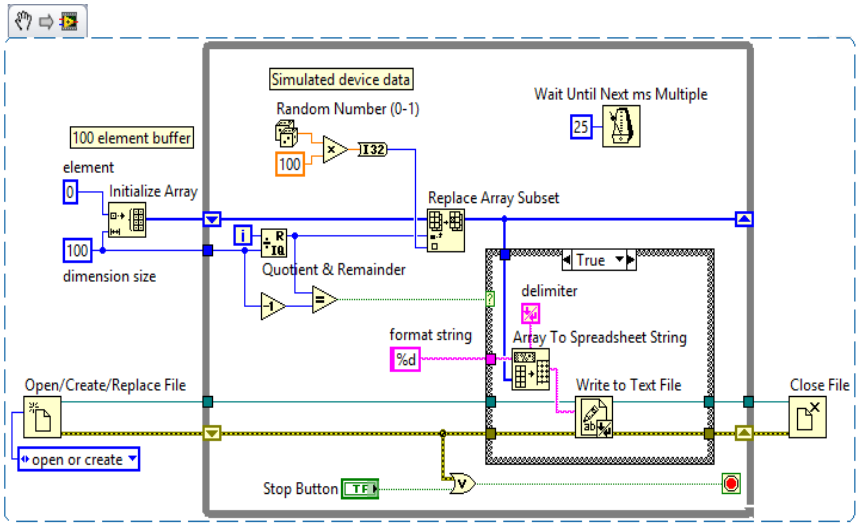

Fig. 1. Example of the program in the LabVIEW environment ("LabVIEW National Instruments", 2018)

3. HTS 2G SUPERCONDUCTING TAPES

The second generation superconducting tapes are made in thin-film technology. They consist of a number of layers from which one can distinguish: the substrate layer, stabilizer layer, buffer layers and the superconductor layer. The structure of the 2G Superconducting tape manufactured by SuperPower is shown in Figure 2.

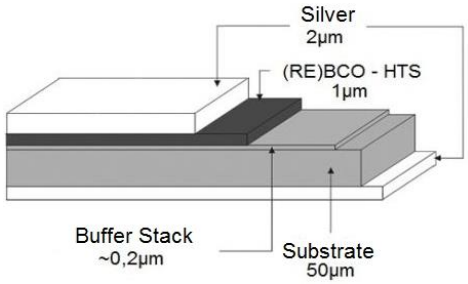

Fig. 2. Structure of HTS 2G superconducting tape without copper stabilizer ("SuperPower Inc.", 2018)

Currently, HTS 2G superconducting tapes are manufactured without stabilizer and with copper stabilizer with 2 mm, 3 mm, 4 mm widths (Fig. 3a), 6 mm and 12 mm (Fig. 3b), substrate thicknesses of 50 µm and 100 µm with a silver layer of 2 µm to 4,5 µm ("SuperPower Inc.", 2018).

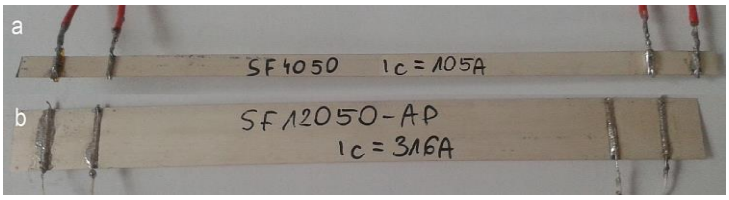

Fig. 3. SuperPower HTS 2G SF4050 superconducting tapes without copper stabilizer with 4mm (a) and 12mm width (b)

The basic parameters of SF type tapes produced by SuperPower are shown in Table 1.

Tab. 1. The parameters of the SF4050 and SF12050 tapes (SuperPower Inc., 2018)

Tape HTS 2G	SF4050	SF12050
Tape Width	4 mm	12 mm
Total Wire Thickness	55 µm	55 µm
HTS YBCO Layer Thickness	1 µm	1 µm
Silver Overlayer Thickness (upper/lower)	2.0/1.8 µm	2.0/1.8 µm
Substrate (Hastelloy C276) Thickness	50 µm	50 µm
Critical Bend Diameter in Tension	11 mm	11 mm
Critical Current $I_{C\,(77,4\,K)}$	105 A	316 A
Critical Temperature T_C	~92 K	~92 K

The HTS 2G superconducting tapes are used in such devices in which traditional conductors are replaced with superconducting equivalents. These include: cables, transformers or motors, and devices that do not have equivalents in the form of conventional conductors. The second group of devices in which the HTS 2G superconducting tapes are used include, among others, superconducting electromagnets, superconducting fault current limiters SFCL, generators (Kozak & Majka, 2014; Naeckel & Noe, 2014; Kozak, Majka, Janowski, Kozak, Wojtasiewicz & Kondratowicz-Kucewicz, 2011).

At operational currents, the current in the superconducting tape is smaller than the critical current I_C of the tape. In the superconductive state, the current flows in the layer of the superconductor YBCO with the omission of other layers. During a fault, the value of the current in the tape exceeds the value of the critical current I_C several times. The superconductor goes to the resistive state and the current in the tape flows mainly through the silver layer (about 80–90%) and the substrate. During a fault, the superconducting tape starts heating up very quickly, which is why the tape should have high resistance in a resistive state and a suitable thermal capacity. During a fault, the temperature and resistance of the tape increases, and therefore the experimental determination of its characteristics R=f(T) is necessary.

4. TEST OF HTS 2G SUPERCONDUCTING TAPES USING THE LabVIEW ENVIRONMENT

Tests of resistance in the temperature function R=f(T) were performed for two types of HTS 2G superconducting tapes: SF4050 and SF12050. The characteristics R=f(T) were determined using the four-wire method using current (I–, I +) and voltage (V–, V +) leads to measure voltages spaced at 100 mm (Fig. 4). The tape sample holder was made of a glass-epoxy composite to which two copper flat bars insulated with 50 μm thick polyimide tape are fixed. After turning the holder, the superconducting tape adheres on both sides to the flat bars; the sample is electrically insulated, and its temperature is measured by the LakeShore Cryotronics CERNOX sensor placed in a hole made in one of the copper flat bars. The test leads are soldered to the superconducting tape sample by means of a hot air soldering tool with the Sn62Pb36Ag2 alloy at 210 ºC.

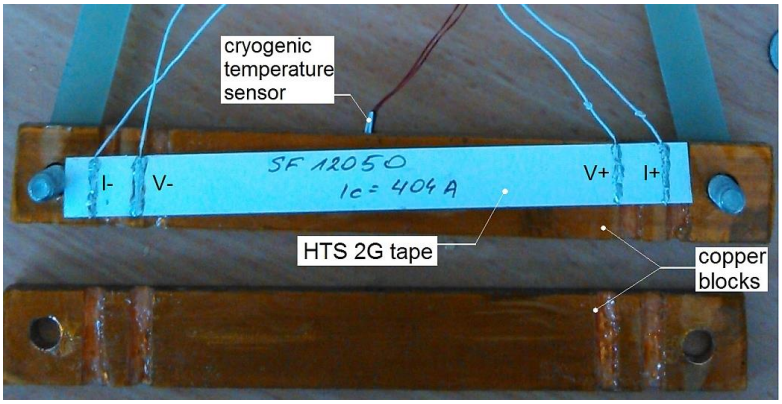

Fig. 4. HTS 2G tape sample in the holder

To carry out measurements and determine the characteristics R=f(T) of HTS 2G SF4050 and SF12050 superconducting tapes, a laboratory measuring system was built (Fig. 4). The measuring system consists of a computer with software written in the LabVIEV environment, the Lake Shore Temperature Monitor 218 temperature meter with a CERNOX cryogenic temperature sensor, NI USB-6343 measuring card, liquid nitrogen cryostat, HTS 2G tape sample holder. Communication between the temperature meter and the computer is carried out via the GPIB port using the GPIB-USB-HS + converter. To measure the current (voltage on the shunt) and the voltage on the tape sample, two analog inputs of the USB-6343 card were used (Fig. 5). Measurement of the superconductor tape resistance is carried out every 0.1 K for 50 ms, in order not to additionally heat the sample with 100 mA.

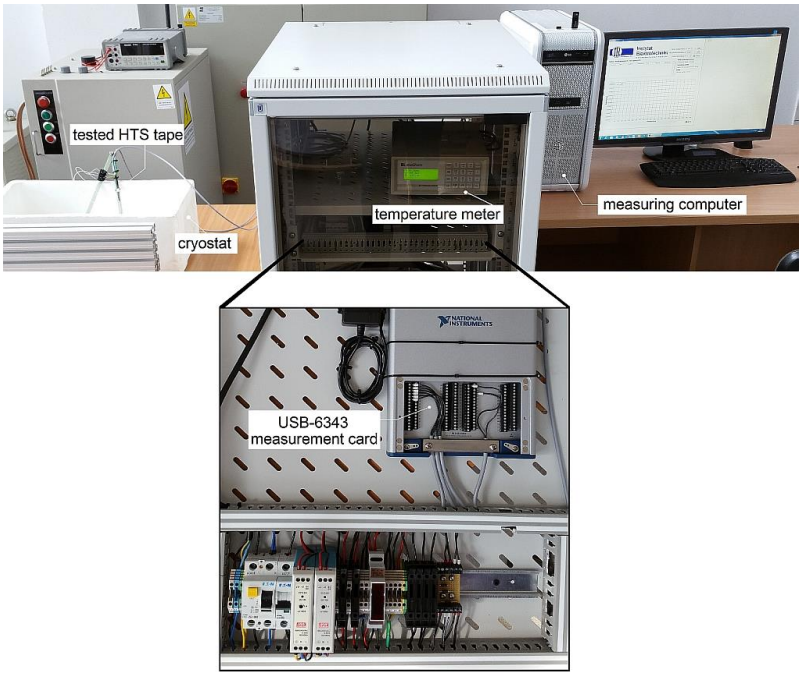

Fig. 5. Laboratory measurement system (own study)

4.1. Test of the characteristics *R*=f(*T*)

Measurement of the characteristics *R*=f(*T*) of the tape was made in the measurement system shown in Figure 5. After cooling the holder with the sample of the superconducting tape in a liquid nitrogen bath to 77.4 K (Fig. 6) the handle was removed from the cryostat.

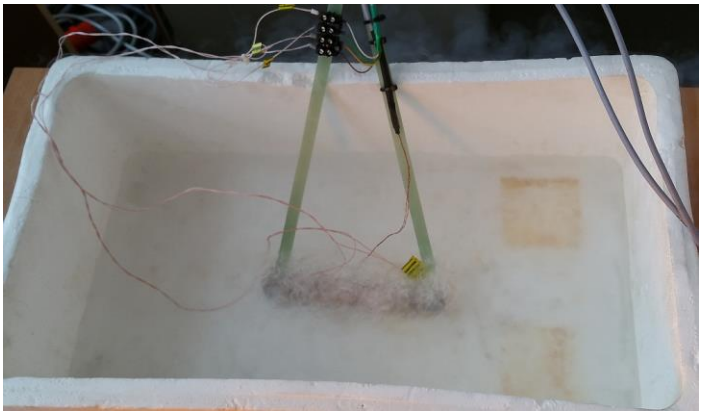

Fig. 6. Sample of HTS 2G tape placed in the holder during cooling in liquid nitrogen

During the slow heating, the results of the measurements are recorded in the temperature range 77 K – 300 K. For the registration of measurements, IEl RT program was written in the LabView environment (Fig. 7). The program allows one to read the temperature from the LakeShore Temperature monitor 218 through the GPIB port, control the relay, and save to * .xls file of all registered parameters.

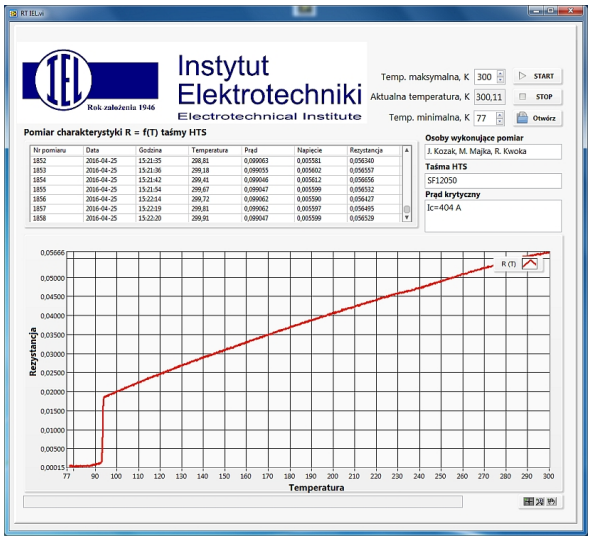

Fig. 7. Determining the characteristics *R*=f(*T*) for HTS 2G SF12050 tape using the LabView environment

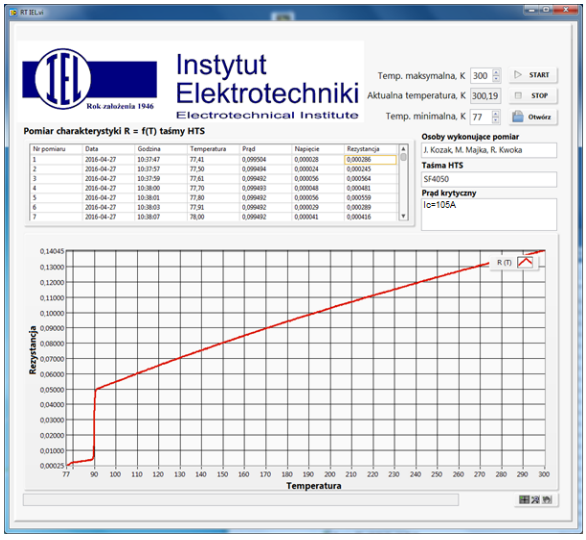

Fig. 8. Determining the characteristics *R*=f(*T*) for HTS 2G SF4050 tape using the LabView environment

The graph (Fig. 9) presents the obtained characteristics R=f(T) of the tested superconducting tapes. The resistance of both tapes measured at 77,4 K is equal to zero.

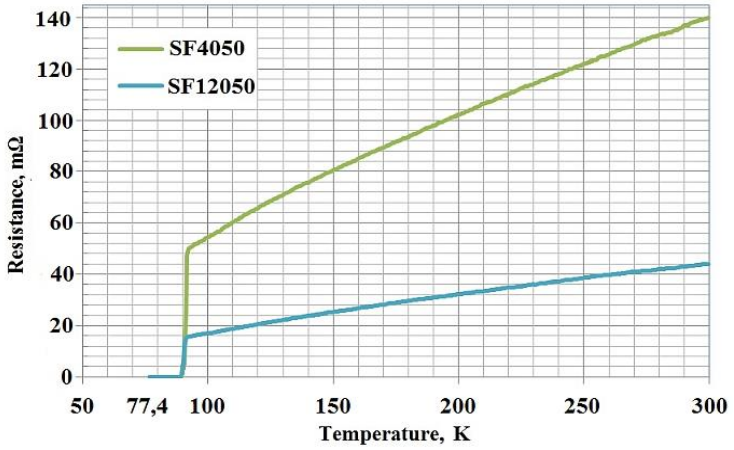

Fig. 9. List of test results concerning characteristics R=f(T)

The graph shows the characteristic transition of the superconductor from the superconductive state to the resistive state at the critical temperature T_C. SF12050 tape has approximately three times lower resistance than SF4050 tape after exceeding T_C. In the calculation of windings in superconducting fault current limiters, both the resistance of the tape and thermal capacity are taken into account. These parameters make it possible to calculate the tape temperature during a fault.

3. CONCLUSIONS

The article presents the experimentally determined characteristics R=f(T) of the second generation superconducting tapes (SF4050 and SF12050) without copper stabilizer. The tests were carried out in the temperature range from 77,4 to 300 K. The article presents a method of preparing samples of superconducting tapes for testing, a measurement method and a way of making measurements. The integrated LabVIEW environment was applied as a tool using the graphical programming language G. Consequently, it was possible to construct virtual devices intended for data collection, processing, analysis, visualization, and for process and measurement control. The experimentally determined characteristics R=f(T) are necessary for the design of superconducting current limiters and the analysis of their numerical models.

REFERENCES

Czerwinski, D., Jaroszynski, L., Majka, M., Kozak, J., & Charmas, B. (2016). Comparison of Over-current Responses of 2G HTS Tapes. *IEEE Transactions on Applied Superconductivity*, *26*(3), 5602904. doi:10.1109/TASC.2016.2520080

Kozak, J., & Majka, M. (2014). Technologia łączenia taśm HTS 2G w uzwojeniach nadprzewodnikowych urządzeń elektrycznych. *Przegląd Elektrotechniczny*, *90*(3), 157–160. doi: 10.12915/pe.2014.03.34

Kozak, J., Majka, M., & Kwoka, R. (2017). Badania taśm nadprzewodnikowych (2G HTS) bez stabilizatora. *Przegląd Elektrotechniczny*, *93*(3), 185-188. doi: 10.15199/48.2017.03.43

Kozak, J., Majka, M., Janowski, T., Kozak, S., Wojtasiewicz, G., & Kondratowicz-Kucewicz, B. (2011). Tests and performance analysis of coreless inductive HTS fault current limiters. *IEEE Transactions on Applied Superconductivity*, *21*(3), 1303–1306. doi: 10.1109/ TASC.2010.2101033

LabVIEW National Instruments. (n.d.). Retrieved January 10, 2018, from LabVIEW National Instruments website, http://www.ni.com/pl-pl.html

Naeckel, O., & Noe, M. (2014). Design and Test of an Air Coil Superconducting Fault Current Limiter Demonstrator, *IEEE Trans. Appl. Supercond.*, *24*(3), 5601605. doi: 10.1109/ TASC.2013.2286294

SuperPower Inc. (n.d.). Retrieved January 8, 2018, from SuperPower Inc. website, http://www.superpower-inc.com/content/2g-hts-wire

Applied Computer Science, vol. 14, no. 1, pp. 73–83
doi:10.23743/acs-2018-07

Submitted: 2018-02-15
Revised: 2018-03-15
Accepted: 2018-03-25

industrial robots, robots programming, AS language, MATLAB

Anna CZARNECKA[*]*, Łukasz SOBASZEK*[*]*, Antoni ŚWIĆ*[*]

2D IMAGE-BASED INDUSTRIAL ROBOT END EFFECTOR TRAJECTORY CONTROL ALGORITHM

Abstract

This paper presents an algorithm for programming an industrial robot's end effector path based on 2D images. The first section gives a brief overview of modern solutions for industrial robot implementation. The next section describes the test set-up and the software used in tests. The work also presents the key elements of the controller algorithm and their operation: 2D image processing with MATLAB software, generating the code for robot control in AS language, and implementation of the produced codes to the Kawasaki RS003N robot.

1. INTRODUCTION

Automation in manufacturing and production processes would be virtually impossible without implementation of robots. Robots have become a common solution in a wide range of industrial processes, and have therefore contributed to boosting the efficiency and flexibility of production, increasing work safety, allowing to achieve high quality of production at higher reliability and lower cost (Hajduk & Koukolová, 2015).

Robots replace human employees in harmful working environments, where highly qualified personnel is scarce. Robots perform a wide range of jobs including: welding, painting, lacquering, adhesive joining or distribution of industrial agents, as well as in auxiliary processes, such as cleaning, polishing or grinding (Hajduk, Jenčik, Jezný, & Vargovčik, 2013).

[*] Lublin University of Technology, Faculty of Mechanical Engineering, Institute of Technological Systems of Information, Nadbystrzycka 36, 20-618 Lublin, tel.: +48 81 538 45 35, l.sobaszek@pollub.pl, a.swic@pollub.pl

In addition to typical applications, robots are implemented to perform other jobs, primarily oriented towards assistance in production. A prominent example of such auxiliary jobs is found in logistics, where the concept of autonomous robots building automated transportation systems has been the subject of much systematic investigation and development, particularly with regard to transporting tools, components and subassemblies. Automated robot are also applied in distribution centres to transport and sort goods ("W fabryce Audi", n.d.). Human-robot interaction at a single workstation is yet another scenario of robot implementation, which is a growing trend in industrial processes (Sobaszek, Gola & Varga, 2016; Sobaszek, Gola & Świć, 2017).

One of the most recent developments in robotics is application of industrial robots as machine tools, performing basic machining jobs. There is a marked trend towards implementing industrial robots (equipped with necessary instrumentation) in machining processes in which the regular CNC machine tools are inadequate, owing to intricate geometry or big size of workpieces (Riexinger Information, n.d.).

This section attempted to give a brief summary of robotic applications in the industry. It becomes apparent that the range of jobs that may be performed by automated robots is constantly increasing. Implementation of robots in industrial processes may require defining the trajectory of robot end effector paths in order to perform the jobs. This process is burdened with considerable time expense and demands various analytical works. Therefore, manufacturers offer different solutions aimed at facilitating and accelerating the process of defining trajectory paths, including employing: teaching through demonstration (Haage, Piperagkas, Papadopoulos, Mariolis, Malec, Bekiroglu, Hedelind & Tzovaras, 2017), neural networks (Jin, Li, Yu & He, 2018), 3D vision systems (Wan, Lu, Wu & Harada, 2017), or machine learning (Xu, Yang, Zhong, Wang & Zhao, 2018). The solution introduced in this paper is a 2D image-based algorithm for controlling an industrial robot, which constitutes an alternative solution to classic programming.

2. RESEARCH OBJECTIVES

We attempted to develop an algorithm that would execute the motion of an industrial robot's end effector along the trajectory defined by means of a 2D image. The proposed solution should improve the work of the robot operator and shorten the programming time. Furthermore, the proposed algorithm would be capable of generating the trajectory path of the robot from the technological documentation (generated with CAD software) and would find application in such processes as: welding, soldering, adhesive joining, trimming or more complex machining.

The algorithm was executed and implemented by means of MATLAB software and used commands derived from the AS programming language. Fig. 1 below presents an overview of the developed algorithm.

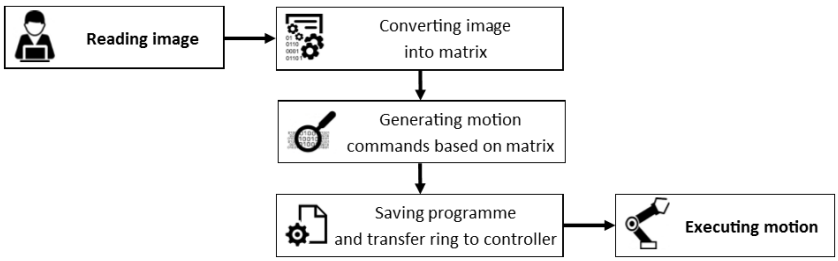

Fig. 1. General overview of the algorithm

3. DESCRIPTION OF THE WORKSTATION

The workstation conditions were simulated by the laboratory test set-up with the Kawasaki RS003N industrial robot. The robot was fitted with an appropriate end effector, a gripper, which enabled verification of the developed solution (Fig. 2).

Fig. 2. Laboratory test-set up with Kawasaki RS003N robot

Kawasaki RS003N robot is a 6-axis industrial robot, programmable with the Teach Pendant, dedicated K-ROSET robot simulation software and AS programming language. The Teach Pendant is an LCD controller that allows to programme and operate the robot, either through execution of AS language commands or by point-to-point teaching. K-ROSET is an advanced programming tool allowing to program robots and perform simulations of robot arm trajectories, analyse cycle-time, check for collision, and verify robot layout. AS is a code developed

by Kawasaki and implemented in the K-ROSET software as a set of commands for programming robot motion, signal handling, gripper operation, etc. (Grobelny, Jarosiński, Piłat, Pieniak & Sobaszek, 2016). The robot motion control is performed by means of the controller, which is additionally responsible for communicating with the robot through special communication protocols. The robot and the controller specifications are shown in Table 1.

Tab. 1. Robot and controller specifications (Kawasaki Heavy Industries, 2010)

Robot specifications	Controller specifications
Model: RS003N-A Type: articulated robot Degree of freedom: 6 Repeatability: ±0,05 mm Maximum load: 3 kg Maximum speed: 6000 mm/s Driving motor: brushless AC servo motor	Model: E70/E71 Structure: enclosed Number of controlled axes: 6 Programming: TEACH/REPEAT Teaching: AS language Power requirements: 200–240 V

4. PROGRAMMING TOOLS

The proposed algorithm was developed and verified with the application of the following programming tools:

- MATLAB software,
- AS programming code.

MATLAB software has a wide range of applications, and may be employed to perform numerical computations, simulations, data visualisation and analysis or image processing. MATLAB is a programming language applied for higher-order programming, and may be applied in works on matrixes, vectors and structures. This software package is applicable in mathematical computations, numerical algorithms, modelling and simulation, data analysis, visualisation of results, engineering graphics and creating functional applications (Fig. 3). In the programming works performed in the study, the following elements of MATLAB package were employed:

- MATLAB scripting language – executed subsequent steps of the algorithm,
- working environment – where the m-files with commands were created and managed,
- selected libraries – performing complex operations on images and matrixes.

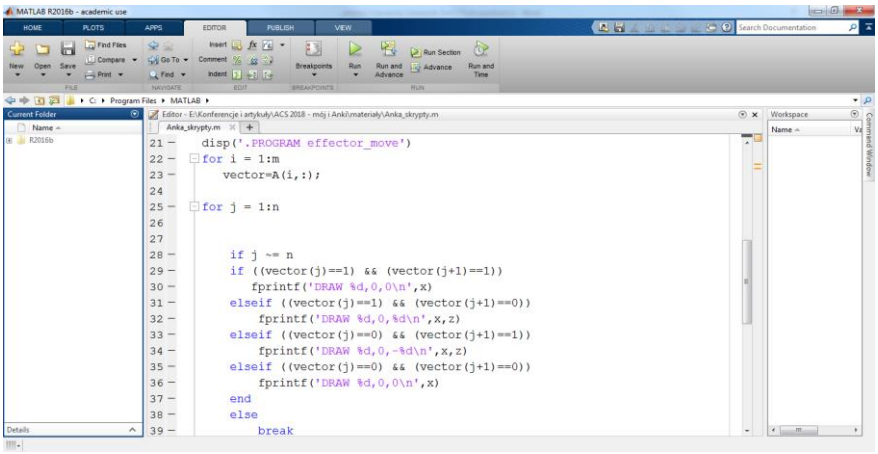

Fig. 3. MATLAB software used in development of algorithm

Programming robots with the AS language resembles classical high-level programming. By means of a series of commands (processed by the controller) the programmer determines the sequence of robot motions. Similarly to other programming languages, the data takes the form of: constant, local variable or global variable. The AS language also makes use of standard data types: numeric expression, combination of alphanumeric characters strings in ASCII, and logical values. The instructions in AS language include commands for programme execution, defining robot locations, motions, velocities, accuracy of robot motion and tool (Kawasaki Heavy Industries, 2010). Fig. 4 shows an example of instructions in AS.

```
.PROGRAM point1()
JMOVE #start
JAPPRO #p001,50
JMOVE #p001
CLAMP 1
.END
```

Fig. 4. A part of code written in AS

The tools used in the research works have been properly integrated, which consisted in generating codes controlling the robot in AS with the application of MATLAB software. The conducted works required employing available image conversion tools, matrix operation instructions, and elements of structural programming. A detailed description of the algorithm will be presented in section 5.

5. DESCRIPTION AND EXECUTION OF ALGORITHM

Each main part of the algorithm, presented in section 2, is composed of a finite number of steps. Each step has its role and functionality. For the purpose of clarity, the developed algorithm is presented in block diagram, whose subsequent steps were executed by means of programming tools (Fig. 7).

First, the algorithm reads the image to be drawn by the robot end effector. Next the image is converted to binary image by means of *im2bw* function. The output bi-level image takes the form of a matrix of elements 0 and 1, which will provide the base for generating the commands of robot motion (Fig. 5).

```
gfx=imread('face.jpg');
A=im2bw(gfx);
[m,n] = size(A);
imshow(A);
```

Fig. 5. Reading and conversion of image

The next stage consists in preparing the environment – the main functionality used here is *diary* function, which allows to save commands generated in AS language as *.pg files. This is a standard file recognised by Kawasaki robots. The subsequent lines of the programme define the values of robot motions (the distance between pixels). This declaration of motion steps enables convenient scaling of the image created by the robot's arm. The distances are specified for each axis (X, Y and Z) (Fig. 6).

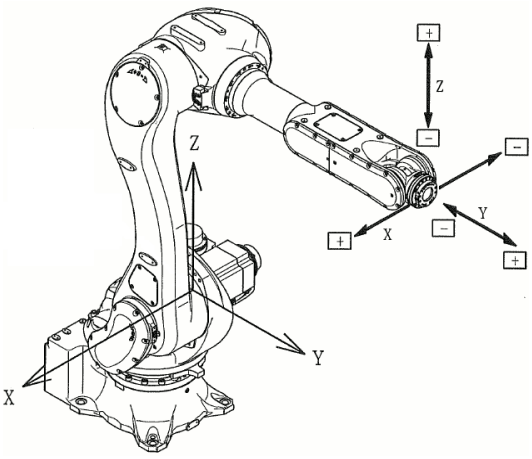

Fig. 6. The robot effector motion options in execution of algorithm (Kawasaki Heavy Industries, 2010)

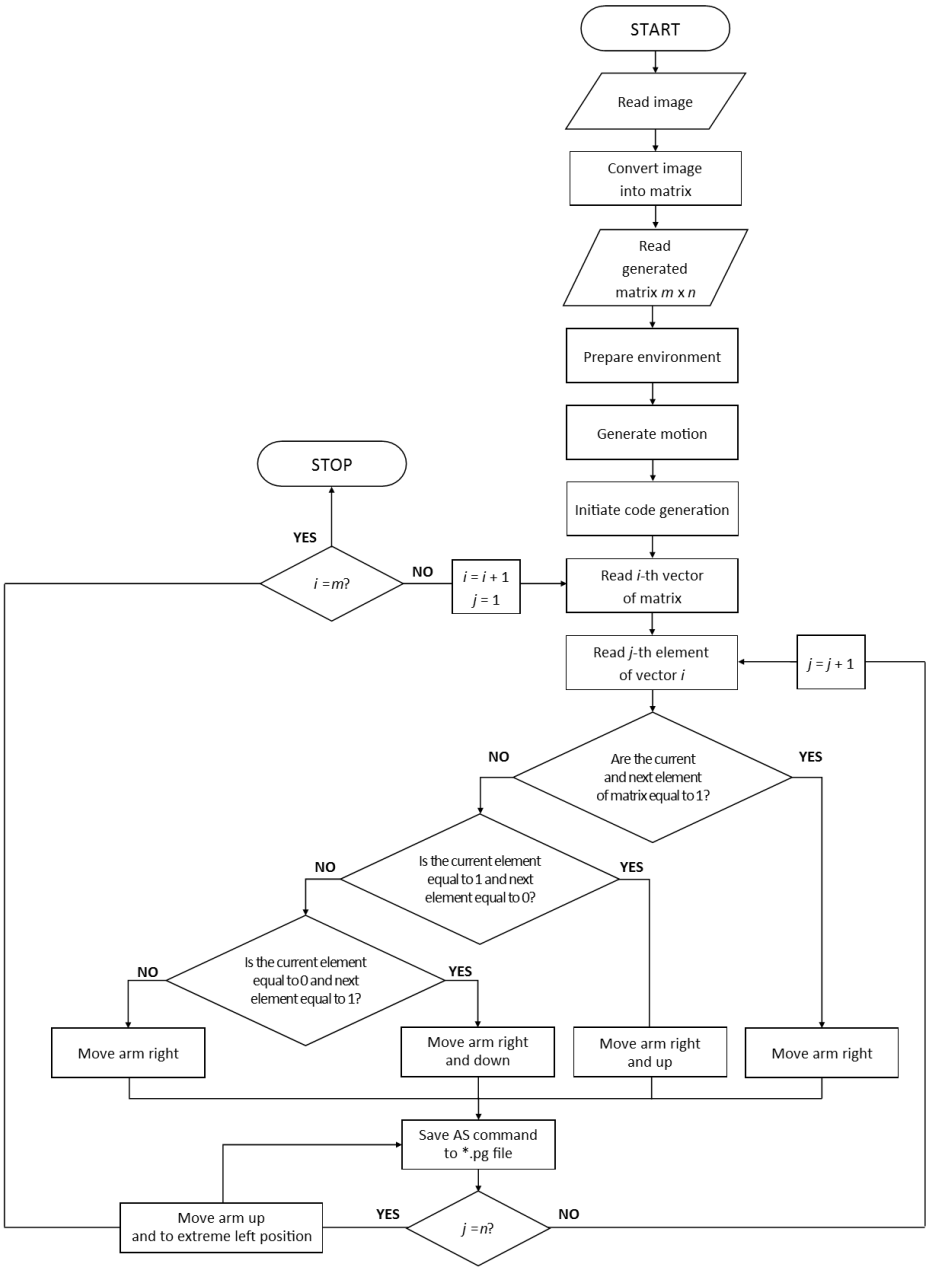

Fig. 7. Block diagram of the Kawasaki industrial robot control algorithm

Once the image is converted and the environment prepared, the fundamental part of the algorithm may commence, *i.e.* generation of robot motion commands. The code is generated iteratively, and therefore includes two *for loops* of specified values. The values depend on the size of the generated matrix (the number of columns and rows). The first loop was responsible for selecting a single matrix row (working vector), and the other one for analysing the values of elements in the working line (Fig. 8).

```
for j = 1:n
if j ~= n
    if ((vector(j)==1) && (vector(j+1)==1))
       fprintf('DRAW %d,0,0\n',X)
    elseif ((vector(j)==1) &&
(vector(j+1)==0))
        fprintf('DRAW %d,0,%d\n',X,Z)
    elseif ((vector(j)==0) &&
(vector(j+1)==1))
        fprintf('DRAW %d,0,-%d\n',X,Z)
    elseif ((vector(j)==0) &&
(vector(j+1)==0))
        fprintf('DRAW %d,0,0\n',X)
    end
else
    break
end
end
```

Fig. 8. Code generating AS instructions – loop analysing elements of the working vector

The analysis of matrix elements consists in checking the current and next element of the matrix. The conditional instruction *if* executes rules that generate instructions in AS language. If the first condition is met, the algorithm generates AS code command, „DRAW X,0,0", which moves the robot right by a previously defined value X. If the condition is not met, subsequent conditions are analysed. The condition which is fulfilled moves the end effector right and up or right and down. The procedure is identical for analysing all elements of the matrix.

The key element of the proposed algorithm is analysing the last element of the working line (Fig. 9). This step is aimed to execute correct pass of the end effector to the first element of the following line. The value of robot end effector motion is determined, and the logical value of the last element of the working line is checked. If the last element is 0, the programme generates instruction „DRAW –X,–Y,0", which causes the robot's arm to move left by *x* and down by *y*, otherwise the generated instruction takes the form of „DRAW 0,0,Z", lifting the arm of the robot by Z, and the instruction „DRAW –X,–Y,0", moving the arm of the robot to the next line.

```
if i ~= m
   Xr=n*X-X;

    if A(i,n)==0
        fprintf('DRAW -%d,-%d,0\n',Xr,Y)
    elseif A(i,n)==1
        fprintf('DRAW 0,0,%d\n',Z)
        fprintf('DRAW -%d,-%d,0\n',Xr,Y)
    end
    if (A(i+1,1)==1)
        fprintf('DRAW 0,0,-%d\n',Z)
    end
    X=x;
else
    break
end
```

Fig. 9. Code determining the pass of the end effector to the subsequent line (vector)

The execution of the algorithm is exported to a *.pg file. After generating, the whole code is sent to the robot's memory by means of KCwinTCP terminal, for remote communication with the robot terminal. The terminal enables executing the code generated by the developed script. Upon completion, the robot end effector motion error verification procedure is possible.

6. VERIFICATION OF ALGORITHM

The developed algorithm was tested in order to verify its efficiency and accuracy. The verification process was twofold:

1. Stage I – draw an image based on a 5x7 test matrix.
2. Stage II – generate end effector trajectory path from a 2D image.

At the first stage the robot was commanded to reflect the paths according to the randomly-generated matrix. In addition, the verification included such areas as the error of motion in X and Y axis and the height Z to which the robot arm was elevated if the matrix values were changed. The results of verification are shown in Fig. 10. Having successfully conducted stage I, stage II of verification was aimed to determine the execution of robot end effector motion based on a 2D image.

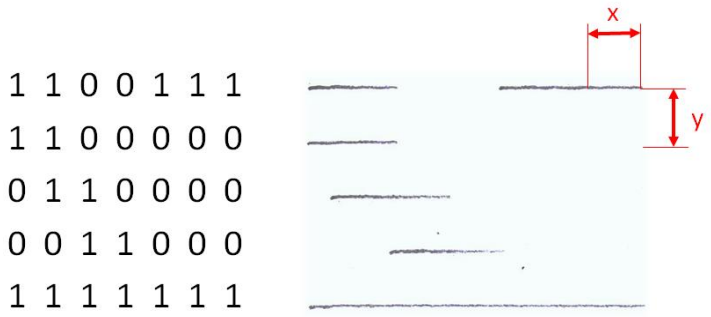

Fig. 10. Model matrix and its execution by the robot

The second stage of verification consisted in analysing the entire algorithm and its execution with the generated code. The selected image was converted into a matrix and drawn by means of the Kawasaki RS003N robot. The results of algorithm execution are shown in Fig. 11.

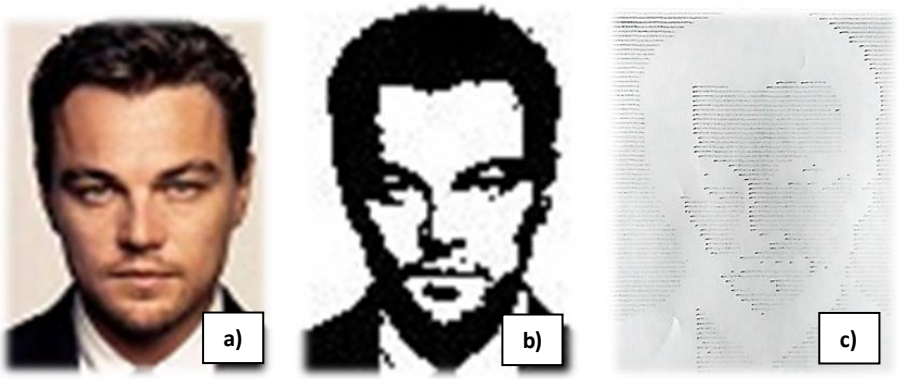

Fig. 11. Verification of image-processing algorithm: a) real image, b) binary image, c) image drawn by the robot – image source: (Sample image, 2014)

Clearly, the proposed algorithm has proven to be functional. The only element of the code that requires further development is the code generation mode, due to the reason that the image is processed into negative. It is, therefore, necessary to implement a special function executing appropriate conversion of the matrix generated from the 2D image.

7. SUMMARY AND FUTURE WORKS

The presented works are in keeping with the current trends towards designing applications for controlling industrial robots and thus implementing robots in applications which require defining end effector trajectory path, which may be of a high degree of complexity. Although the algorithm is operational, future works should concentrate on improving the script-generating code in order to enhance image processing, and, secondly, to generate commands controlling the motion in the vertical plane and along curves. Further works should be carried out in order to enable implementing images made in popular CAD tools. That would open the gate to applying the algorithm to such processes as trimming, soldering or adhesive joining.

REFERENCES

Grobelny, M., Jarosiński, S., Piłat, M., Pieniak, D., & Sobaszek, Ł. (2016). Projekt aplikacji komputerowej umożliwiającej sterowanie robotem przemysłowym kawasaki RS003N. *Zeszyty Naukowe Wydziału Elektroniki i Informatyki*, 10, 163–176.

Haage, M., Piperagkas, G., Papadopoulos C., Mariolis I., Malec J., Bekiroglu Y., Hedelind M., & Tzovaras D. (2017). Teaching Assembly by Demonstration Using Advanced Human Robot Interaction and a Knowledge Integration Framework. *Procedia Manufacturing*, 11, 164–173. doi:10.1016/j.promfg.2017.07.221

Hajduk M., & Koukolová L. (2015). Trends in Industrial and Service Robot Application. *Applied Mechanics and Materials*, 791, 161–165. doi:10.4028/www.scientific.net/AMM.791.161

Hajduk, M., Jenčik, P., Jezný, J., & Vargovčik, L. (2013). Trends in industrial robotics development. *Applied Mechanics and Materials*, 282, 1–6. doi:10.4028/www.scientific.net/AMM.282.1

Jin L., Li, S., Yu, J. & He, J. (2018), Robot manipulator control using neural networks: A survey. Neurocomputing, 285, 23–34. doi:10.1016/j.neucom.2018.01.002

Kawasaki Heavy Industries. (2010). *AS Language Programming*.

Kawasaki Heavy Industries. (2010). *Kawasaki Robots User Manual*.

Riexinger Information. (n.d.). Retrieved May 20, 2017, from https://riex.de/automatisierung/roboterfraesanlage

Sample image [online image]. (2014). Retrieved May 20, 2017, from http://www.boomsbeat.com/articles/1875/20140327

Sobaszek, Ł., Gola, A., & Świć A. (2017). Kierunki rozwoju robotyki w aspekcie projektowania współczesnych systemów produkcyjnych. In R. Knosala (Eds.), *Innowacje w zarządzaniu i inżynierii produkcji* (pp. 460–471). Opole: Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją.

Sobaszek, Ł., Gola, A., & Varga, J. (2016). Virtual designing of robotic workstations. *Applied Mechanics and Materials*, 844, 31–37. doi:10.4028/www.scientific.net/AMM.844.31

W fabryce Audi roboty transportują samochody. (n.d.). Retrieved March 15, 2018, from Audi Autorund Website, http://autorud.pl/audiblog/w-fabryce-audi-roboty-transportuja-samochody

Wan, W., Lu, F., Wu, Z., & Harada, K. (2017). Teaching robots to do object assembly using multi-modal 3D vision. *Neurocomputing*, 259, 85–93. doi:/10.1016/j.neucom.2017.01.077.

Xu, Y., Yang, Ch., Zhong, J., Wang, N., & Zhao, L. (2018). Robot teaching by teleoperation based on visual interaction and extreme learning machine. *Neurocomputing*, 275, 2093–2103. doi:10.1016/j.neucom.2017.10.034

Applied Computer Science, vol. 14, no. 1, pp. 84–93
doi:10.23743/acs-2018-08

Submitted: 2018-03-01
Revised: 2018-03-22
Accepted: 2018-03-26

computed tomography, rapid prototyping, reverse engineering

Robert KARPIŃSKI[*], *Józef JONAK*[*], *Jacek MAKSYMIUK*[**]

MEDICAL IMAGING AND 3D RECONSTRUCTION FOR OBTAINING THE GEOMETRICAL AND PHYSICAL MODEL OF A CONGENITAL BILATERAL RADIO-ULNAR SYNOSTOSIS

Abstract

The paper presents results of a 3D reconstruction of a congenital bilateral radio-ulnar synostosis. Basics of anatomy and biomechanical analysis of the elbow joint were introduced. Case report of a congenital bilateral radio-ulnar synostosis was presented. Based on the data from computed tomography imaging, the model of a congenital bilateral radio-ulnar synostosis was constructed. Basic information on reverse engineering, rapid prototyping and methods of making physical models are presented. The creation of physical models was aimed at pre-operative planning and conceptualization. Physical models were also used in the educational form at the stage of communication with the patient.

1. INTRODUCTION

Skeletal diseases and injuries are considered as one of the most important problems of modern civilization. The combination of potential technical and medical sciences has enabled the development of biomedical engineering, especially in the field of medical imaging techniques and typical issues in the field of engineering modelling. The effect of these activities is the ability to obtain three-dimensional models, both virtual and physical, of any anatomical structures of living organisms (Skalski & Haraburda, 2009; Karpiński, Jaworski & Zubrzycki, 2016; Kozłowska & Zubrzycki, 2017).

[*] Department of Machine Design and Mechatronics, Faculty of Mechanical Engineering, Lublin University of Technology, Nadbystrzycka 36, 20-618 Lublin, Poland, r.karpinski@pollub.pl
[**] Orthopedic Department, Łęczna Hospital, Krasnystawska 52, 21-010 Łęczna, Poland

The use of reverse engineering and rapid prototyping methods is extremely important at the stage of planning of the surgical procedure and plays a significant role in educational process (Holubar, Hassinger & Dozois, 2009; Hurson, Tansey, O'Donnchadha, Nicholson, Rice & McElwain, 2007; McGurk, Potamianos, Amis & Goodger, 1997; Petzold, Zeilhofer & Kalender, 1999). Rapid Prototyping (RP) is a production technology that is currently used in many industries. It is used to create three-dimensional models accurately reproducing structures based on source data from imaging, and then to make their physical counterparts (Zubrzycki & Braniewska, 2017; Frame & Huntley, 2012). In orthopaedics and traumatology of the locomotor system this technique finds application in cases where traditional imaging methods are insufficient, especially in cases where complicated fractures and deformations occur, in which it is necessary to perform a 3D reconstruction (Brown, Firoozbakhsh, DeCoster, Reyna Jr, & Moneim, 2003; Guarino, Tennyson, McCain, Bond, Shea & King, 2007).

The digital record of the results of medical imaging makes it possible to present the reverse engineering process, in which the real model is transformed into a virtual model using the spatial measurement technique. This technique plays a huge role in clinical practice, especially at the stage of planning the surgical procedure and determining its course. The created models are also used as an educational tool. Visualization of anatomical structures through the creation of virtual and material models gives the opportunity to analyze the musculoskeletal structures, classify and assess clinical cases and consider possible treatment options (Mulford, Babazadeh & Mackay, 2016; Bagaria, Deshpande, Rasalkar, Kuthe & Paunipagar, 2011).

In the described case, reconstruction of bone structure congenital bilateral radio-ulnar synostosis on both upper limbs was performed on the basis of image data from computed tomography. The creation of physical models was aimed at pre-operative planning and conceptualization, an attempt to separate the connected bones. Physical models were also used in the educational form at the stage of communication with the patient.

Written informed consent was obtained from the patient for publication of this case report and any accompanying images.

2. ANATOMY OF THE ELBOW JOINT

Human have reached a high stage in his evolutionary development, among other things thanks to the excellent functions of the upper limbs. It was possible because of the complicated structure, which consists of numerous joints, muscles and bones as well as vessels and nerves. The elbow joint and proper connection between the bones of the forearm enables a proper positioning of the hand for precise work.

The elbow joint is a hinge joint that enables flexion and extension of the forearm. It consists of the brachio-ulnar and brachioral-radial connections that fully support the lateral collateral and medial collateral ligaments. In the proper functioning of the entire upper limb, the proximal radio-ulnar joint plays a significant role. This is a rotary joint in which movements are pronation and supination of the forearm takes place between the head of the radius and the radial notch of the ulna. Stems of the radius and ulna bones form the central radial-cubital joint. Interosseous membrane connecting these two bones by a 3 centimetres distal to the tuberosity of the radius bone and extend over the entire length of the forearm. The distal radio-ulnar joint is a rotating joint formed by the head of the ulna and ulnar indentation of the radius bone. Movements performed in this joint are conversion (pronation) and inverting (supination) of the forearm.

Congenital radio-ulnar synostosis is a rarely occurring anomaly. The case was firstly described by Sandifort in 1797. To this day only 350 cases have been recorded. Its inheritance may be autosomal and dominant (Rizzo, Pavone, Corsello, Sorge & Opitz, 1997).

During the embryonic development, at approximately 5th week, cartilaginous analogues of radial, ulnar and humeral bones are unified for a short time. At about 8th week, forearm bones are placed in pronation, and in this position an ossification begins. If a correct division of ulnar and radial bone doesn't occur, they remain connected. In consequence, the synostosis limits the movements of pronation and supination, what could be noted approximately three years after birth, when the child demonstrates problems with catching a ball and holds toys with the hand directed to the back (Hansen & Andersen, 1970). Congenital radio-ulnar synostosis is bilateral in 60% of cases and occurs at men more often than at women in proportion of 3:2 (Al-Saadi & Havekrog, 2008).

3. CONGENITAL BILATERAL RADIO-ULNAR SYNOSTOSIS CASE REPORT

The paper focuses on a case of a 30-year-old Caucasian female with bilateral congenital radio-ulnar synostosis in proximal parts of both forearms, without other changes or inborn diseases. The patient was diagnosed with double congenital radio-ulnar synostosis 6 months after birth. It was decided to start an observation period of the patient, without any surgical interventions. At the age of 2, limitations in pronation and supination movements of both forearms had been noticed. It manifested with difficulties while playing with other children, holding a pen and performing household activities. After 8 years, a faulty posture was diagnosed, therefore the attending physician recommended exercises for strengthening back muscles as well as respiratory exercises. At the age of 25, the patient started to complain about pain in her right limb, therefore a rehabilitation

and pain relievers were prescribed. Five years later the carpal tunnel syndrome of the right wrist was diagnosed, and a year after that the carpal tunnel release surgery was performed, reducing the pain. However, in this period the patient started observing numbness and cramps in left hand, mainly while writing and combing. An ultrasound scan of both forearms was conducted, resulting in the manifestation of a conflict of tendon and median nerve. A comparative X-ray test was carried out on both forearms with an elbow joint in the postero-anterior (Fig. 1a) and lateral projection (Fig. 1b), illustrating changes typical for the bi-lateral congenital radio-ulnar synostosis in proximal part of the limb.

Due to persistent pain and numbness complaints of the left hand, magnetic resonance was advised for both wrists, however no changes were stated within median nerves.

At the age of 32, the patient reported the pain in the right hand, hence the idea of performing the electromyography (EMG) exam, which was conducted on me-dian, radial and ulnar nerves. In the examination of the ulnar nerve, a double – sided decrease of amplitude were stated, approximately 2 cm below cubital tunnel, with areas indicating a blockage of conductivity.

In order to assess the exact anatomical changes, a computed tomography (CT) with 3D reconstruction was advised for both forearms (including elbow joints).

Fig. 1. X-ray image of the left forearm in postero-anterior (a) and lateral (b) projection

4. MEDICAL IMAGING AND 3D RECONSTRUCTION

A computed tomography is a diagnostic method used for non-invasive spatial imaging of patient's internal organs by conducting a sequence of body-section X-Ray images.

Materialise Mimics software allows user to process and edit 2D DICOM images obtained from computed tomography or magnetic resonance imaging scanners in order to generate 3D models. Software can distinguish several structures and tissues of human body, based on their colour in the Hounsfield scale, reflecting an apparent density of discussed tissue(Karpiński, Jaworski & Zubrzycki, 2017; Karpiński, Jaworski, Szala & Mańko, 2017).

If needed, a value of the apparent density can be assigned to a corresponding area of a 3D model in order to obtain a more accurate model for further research, involving finite elements analysis in particular. Furthermore, researchers can evaluate and improve the geometrical model using Remesh tool, e.g. to reduce the number of faces in a polygon mesh or to smooth the surface of the model (Zubrzycki, Karpiński & Górniak, 2016).

4.1. Background on computed tomography

X-ray transmission computer tomography is a type of X-ray spectroscopy. Computed tomography is a non-invasive diagnostic method that allows obtaining layer images of the examined object. It allows the imaging of the spatial distribution of the internal organs of the patient. The test consists of performing the sequence of layered X-ray images in a plane perpendicular to the axis of the body. In computed tomography, a thin transverse cross-section of the body is irradiated at many angles with a narrow X-ray beam, which after passing through the examined object is registered on its other side on the detector panel. X-rays passing through the test object are weakened, which is described as a function of energy of radiation, thickness and a type of the test material. The amount of the beam that has been dispersed or absorbed per unit of thickness of the object exposed to radiation is represented by the linear absorption coefficient. The radiation passing through the tissues is measured with a scintillation counter. The collected data is introduced to computer software where, with the use of appropriate mathematical algorithms, its analysis and reconstruction is performed in the form of an image of the screened layer. Collected radiographic images represent the difference depending on the absorption coefficient of the radiation for particular tissues. Computer tomography images are the most popular source of data on the geometry of human internal structures. When the projection is gathered information about the apparent density of each layer of radiation projection by measuring the coefficient of HU (Hounsfield) (Skalski, Grygoruk, Makuch & Dąbrowska-Tkaczyk, 2015; Ratajczyk, 2012; Budzik, Dziubek & Turek, 2015).

4.2. Developing of the virtual model of the structure

A series of computed tomography scans was conducted, resulting in a set of 2D images of consecutive cross-sections of studied limb, perpendicular to the long axis of the bone. These images were then exported to DICOM file format and implemented in Materialise Mimics software. Based on source files the program calculates cross-sections in two perpendicular planes parallel to the long axis of the bone and the user assigns a frame of reference to each of the three planes.

The process of generating the 3D model starts with applying a filter called Threshold, which highlights tissues on images with a value of radiodensity within specified range. For bones the threshold usually displays areas with density between 226 and 1895 in Hounsfield scale. Next, one can divide the selection in several different masks using Region Growing, which can be helpful in cases of complex structures, when it is convenient to switch between models of parts of the setup, e.g. bones with corresponding joints etc. When the mask of desired tissue is ready, the user orders a process of calculating the 3D model based on the highlighted tissues on layers in three perpendicular planes of the 3D coordinate system (Fig. 2).

The quality of the model is mostly dependent on two factors: a precision of the user in operating on layers and the quality of images themselves. In the discussed case the imperfections of the scan set are noticeable, probably as a result of a resolution of the device and a distance between consecutive layers of scanning, leading to local blurs and softening of the edges. These phenomena resulted in an inability to automatically distinguish wanted bones, hence the necessity to manually separate discussed structures, using tools for mask editing layer by layer or in 3D point cloud and algorithms for Boolean operations on the masks.

Once the model is calculated, it can be processed using 3-maticsoftware tools to obtain a surface of higher quality. Its main function is to apply a polygon mesh on the model, usually composed of triangular faces, and then to assess and modify its properties. Based on the given structure, the mesh could be comprised of varying number of triangles. If further analysesare required, e.g. finite elements analysis, it is crucial to reduce the number of faces to lowest possible value, which could shorten a time required for appropriate analysis (Fig. 3).

Due to inadequacy of the model compared to real structures, which occurred because of the insufficient quality of provided images, it had to be refined. First of all, any sharp edges had to be smoothed to obtain a quality of surface as close to the surface of the corresponding bones as possible. This action resulted in unwanted increase of triangles in the mesh, thus the next step was to rearrange the mesh grid and reduce the number of triangles with preservation of the surfaces. Once the mesh was simplified, the model was ready for exporting to any desired software, including ABAQUS and Comsol for further analyses or as a STEP of STL file for special processing in CAD software.

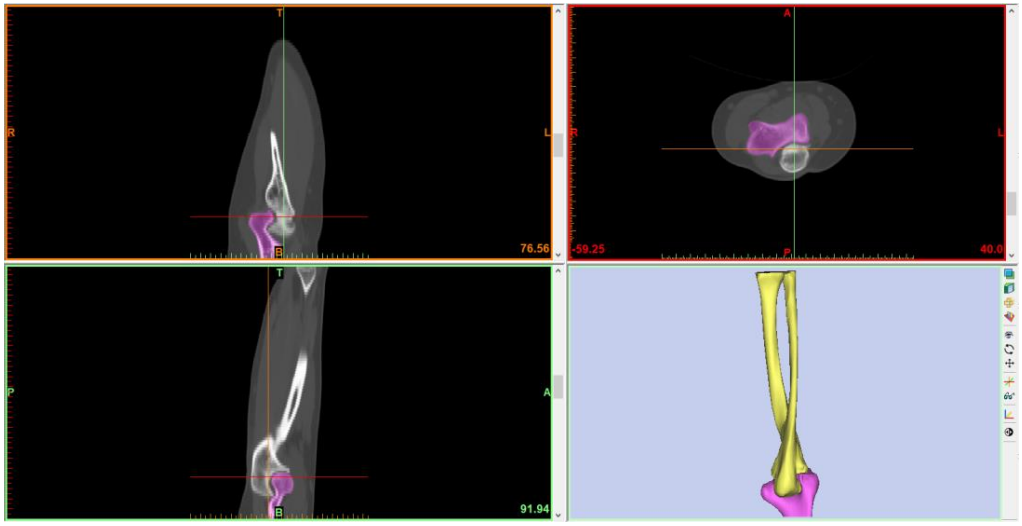

Fig. 2. The layers and a preliminary 3D model of an arm

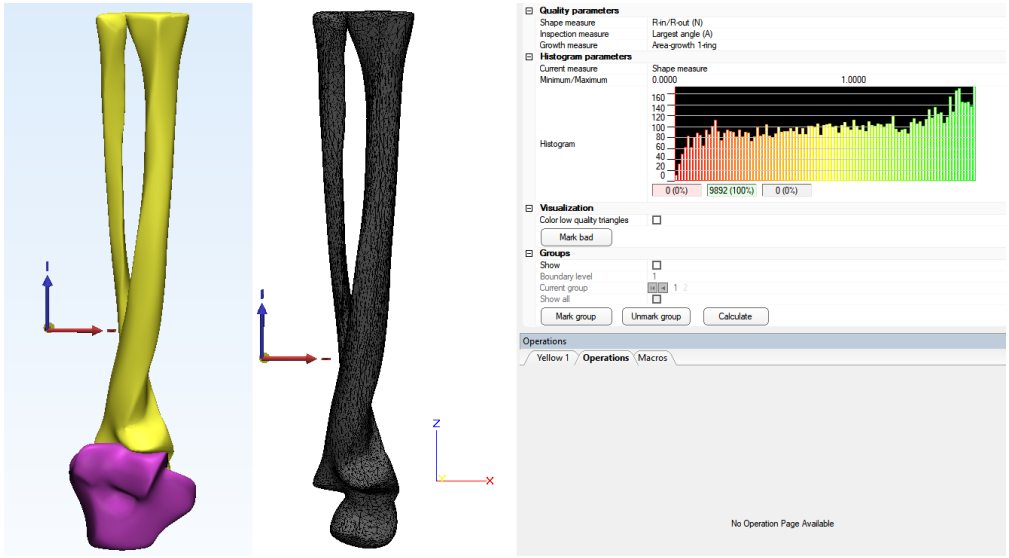

Fig. 3. The model in 3-matic software after smoothing (left) and triangle reducing with a histogram of the mesh (center and right)

4.3. The manufacturing of the physical model

In presented case, at a stage of planning the procedure of surgically dividing bones in the congenital radio-ulnar synostosis, physical models of mentioned condition were used. These models were created on 3D printer using Fused Deposition Modelling technology (FSD) by extruding a thermoplastic filament, layer by layer, on a moving platform, lowered after each layer. After the process, any residues of structural filament, used as support for regular filament, are being removed, either mechanically or under pressure in a washer.

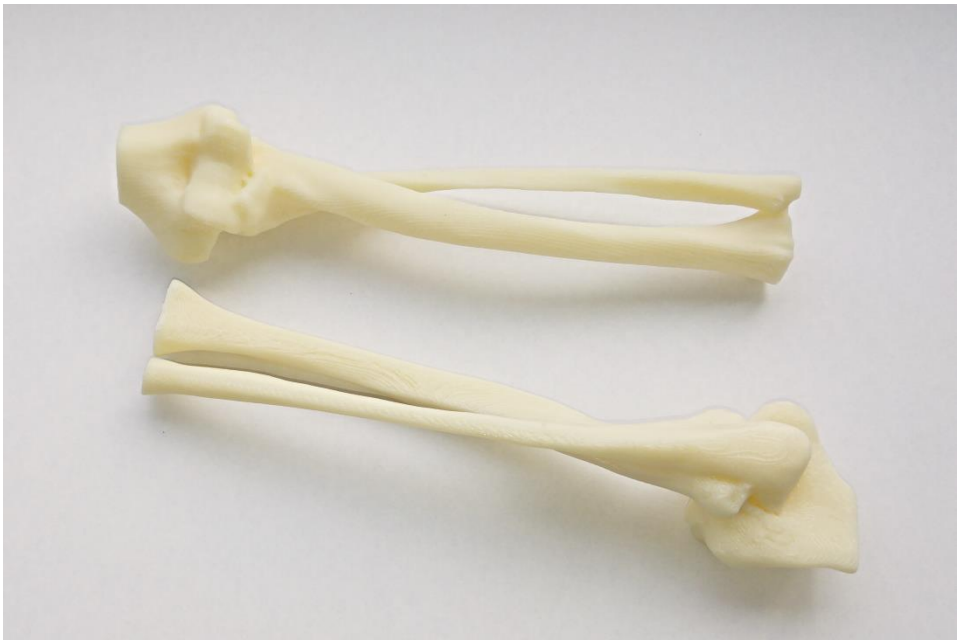

Fig. 4. Physical models of a congenital bilateral radio-ulnar synostosis made by 3D printing method

The input models for 3D printing were represented by triangular mesh grid and saved in STL file format (Stereolithography Interface Specification). They were than manufactured uPrint SE Plus 3D printerusing high impact ABS (Acrylonitrile butadiene styrene). Models presented on Fig. 4 were made of thermoplastic filament of diameter 1.75 mm, and a thickness of each layer was 0.2 mm.

5. CONCLUSIONS

The combination of modern methods of medical imaging with the use of virtual engineering modelling allows very accurate representation of the anatomical structures of living organisms. The implementation of virtual models of damaged or afflicted skeletal components, and then the creation of their physical representations is very important at the planning stage of the surgical procedure, during the selection of the necessary instruments and during communication with the patient. In described case of the congenital bilateral radio-ulnar synostosis, the implementation of physical models allowed for individual planning of the separation of connected bones. Earlier preparation for surgery can significantly shorten the duration and reduce the risk of error which in turn reduces its cost as well as the often possible to reduce the degree of surgical intervention.

REFERENCES

Al-Saadi, Z. S., & Havekrog, B. H. (2008). Congenital radioulnar synostosis. *Ugeskr Laeger, 170*(40), 3147–3148.

Bagaria, V., Deshpande, S., Rasalkar, D. D., Kuthe, A., & Paunipagar, B. K. (2011). Use of rapid prototyping and three-dimensional reconstruction modeling in the management of complex fractures. *European journal of radiology*, *80*(3), 814–820.

Brown, G. A., Firoozbakhsh, K., DeCoster, T. A., Reyna Jr, J. R., & Moneim, M. (2003). Rapid prototyping: the future of trauma surgery? *The Journal of Bone and Joint Surgery*, *85*, 49–55.

Budzik, G., Dziubek, T., & Turek, P. (2015). Podstawowe czynniki wpływające na jakość obrazów tomograficznych. *Problemy Nauk Stosowanych*, *3*, 077–084.

Fahlstrom, S. (1932). Radio-ulnar synostosis: historical review and case report. *The Journal of Bone and Joint Surgery*, *14*(2), 395–403.

Frame, M., & Huntley, J. S. (2012). Rapid prototyping in orthopaedic surgery: a user's guide. *The Scientific World Journal*, *2012*, 838575. doi:10.1100/2012/838575

Guarino, J., Tennyson, S., McCain, G., Bond, L., Shea, K., & King, H. (2007). Rapid prototyping technology for surgeries of the pediatric spine and pelvis: benefits analysis. *Journal of Pediatric Orthopaedics*, 27(8), 955–960.

Hansen, O. H., & Andersen, N. O. (1970). Congenital radio-ulnar synostosis. Report of 37 cases. *Acta Orthopaedica Scandinavica*, *41*(3), 225–230.

Holubar, S. D., Hassinger, J. P., & Dozois, E. J. (2009). Virtual pelvic anatomy and surgery simulator: an innovative tool for teaching pelvic surgical anatomy. *Studies in Health Technology and Informatics*, *142*, 122–124.

Hurson, C., Tansey, A., O'Donnchadha, B., Nicholson, P., Rice, J., & McElwain, J. (2007). Rapid prototyping in the assessment, classification and preoperative planning of acetabular fractures. *Injury*, *38*(10), 1158–1162.

Karpiński, R., Jaworski, Ł., & Zubrzycki, J. (2016). Structural analysis of articular cartilage of the hip joint using finite element method. *Advances in Science and Technology Research Journal*, *10*(31), 240–246. doi:10.12913/22998624/64064

Karpiński, R., Jaworski, Ł., & Zubrzycki, J. (2017). The design and structural analysis of the endoprosthesis of the shoulder joint. *ITM Web of Conferences*, *15*, 07015. doi:10.1051/itmconf/20171507015

Karpiński, R., Jaworski, Ł., Szala, M., & Mańko, M. (2017). Influence of patient position and implant material on the stress distribution in an artificial intervertebral disc of the lumbar vertebrae. *ITM Web of Conferences*, *15*, 07006. doi:10.1051/itmconf/20171507006

Kozłowska, E., & Zubrzycki, J. (2017). Using methods of the reverse engineering to carry personalised preoperative stabilisers out on the example of vertebrae of human spine. *ITM Web of Conferences*, *15*, 02007. doi:10.1051/itmconf/20171502007

McGurk, M., Potamianos, P., Amis, A. A., & Goodger, N. M. (1997). Rapid prototyping techniques for anatomical modelling in medicine. *Annals of the Royal College of Surgeons of England*, *79*(3), 169–174.

Mulford, J. S., Babazadeh, S., & Mackay, N. (2016). Three dimensional printing in orthopaedic surgery: review of current and future applications. *ANZ Journal of Surgery*, *86*(9), 648–653.

Petzold, R., Zeilhofer, H. F., & Kalender, W. A. (1999). Rapid prototyping technology in medicine – basics and applications. *Computerized Medical Imaging and Graphics*, *23*(5), 277–284.

Ratajczyk, E. (2012). Rentgenowska tomografia komputerowa (CT) do zadań przemysłowych. *Pomiary Automatyka Robotyka*, *16*, 104–113.

Rizzo, R., Pavone, V., Corsello, G., Sorge, G., & Opitz, J. M. (1997). Autosomal dominant and sporadic radio-ulnar synostosis. *American Journal of Medical Genetics*, *68*(2), 127–134.

Skalski, K., Grygoruk, R., Makuch, A., & Dąbrowska-Tkaczyk, A. (2015). Modelowanie wirtualne i materialne na potrzeby komputerowego wspomagania zabiegów operacyjnych. In M. Gzik, M. Pawlikowski, M. Lewandowska-Szumieł & M. Wychowański (Eds.), *Biomechanika i Inżynieria Biomedyczna* (pp. 481–498). Akademicka Oficyna Wydawnicza EXIT.

Skalski, K., & Haraburda, M. (2009). *Komputerowe wspomagane projektowanie i wytwarzanie implantów stawów człowieka*. Monografia nr 137. Radom: Wydawnictwo Politechniki Radomskiej.

Zubrzycki, J., & Braniewska, M. (2017). Zastosowanie inżynierii odwrotnej w projektowaniu spersonalizowanego implantu stawu biodrowego. *Mechanik*, 90(1), 46–47.

Zubrzycki, J., Karpiński, R., & Górniak, B. (2016). Computer aided design and structural analysis of the endoprosthesis of the knee joint. *Applied Computer Science*, *12*(2), 84–95.