ДОДАТОК А

Висвітлення результатів у публікаціях

JOURNAL OF UNIVERSAL SCIENCE RESEARCH

OUR FIELDS

- Exact Sciences
- Natural sciences
- Medical sciences
- Technical sciences
- Economics
- Philological sciences
- Pedagogical sciences
- Social and humanitarian sciences
- Psychological sciences
- Arts and cultural sciences
- Physical education and sports

RESEARCHBIB IMPACT
FACTOR: 6.4 / 2023
SJIF 2023: 3.778

Research Science and
Innovation House

OUR INDEXING

ISSN (E): 2181-4570

[illegible]

JOURNAL OF UNIVERSAL SCIENCE RESEARCH

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11

SIGNALS COLLISIONS DETECTION IN WIRELESS NETWORKS

Dmytro Buts, Olena Chala, Svitlana Maksymova

Department of Computer-Integrated Technologies, Automation and Robotics,
Kharkiv National University of Radio Electronics, Ukraine

Abstract:

Within the framework of the concept of Industry and 4.0, including IoT, the use of various sensors is increasingly widespread. It is necessary to understand that it is almost impossible to avoid so-called collisions (that is, overlaps) of signals. Such overlaps can and do lead to errors that can be quite significant. This article discusses the solution to the problem of identifying collisions, and also proposes an algorithm for avoiding such overlaps. The consequence of using this algorithm is to increase the noise immunity of the wireless network.

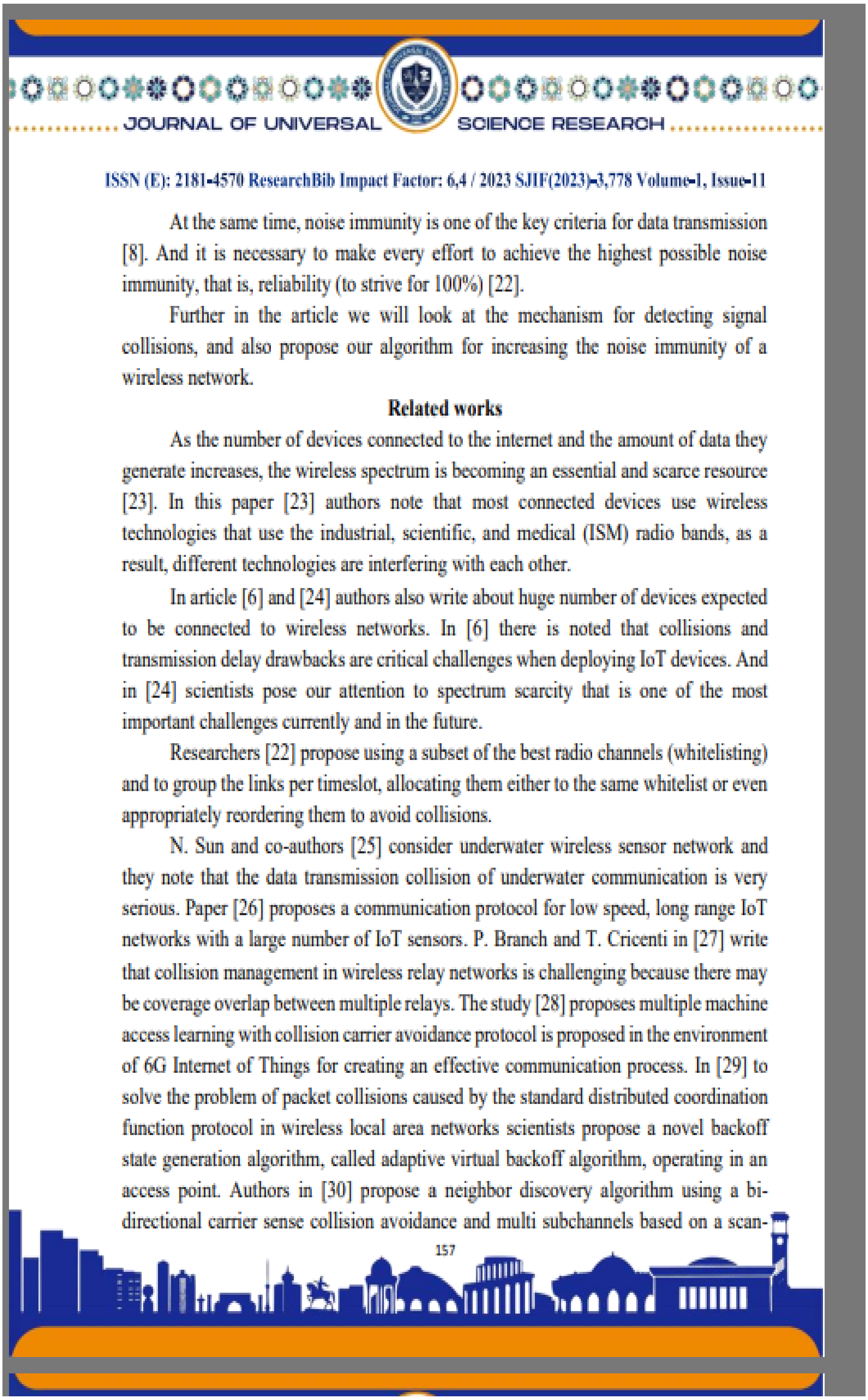
Key words: Industry 4.0, Wireless network, Collision, Node-neighbor, Coordinator node

Introduction

With the spread of the Industry 4.0 concept using, there was a need to improve and develop industrial interfaces and data transfer protocols of integrated systems for automated control [1]-[6]. The predominant choice of wireless data transmission has become quite natural. But here new challenges appear [7]-[21].

The problems of reducing energy consumption in wireless networks and high-reliability (close to 100%) [22] is one of the most important when designing networks whose nodes use autonomous energy sources. The most important problem in such networks is the need to prevent collisions (conflicts).

Collision of signals is their mutual superimposition, which occurs during the simultaneous transmission of signals from two or more sending nodes in a separate network environment, which leads to their distortion, loss of transmitted information, the need for retransmission of data, and, ultimately, to an increase in power consumption of a network node with an autonomous energy source and reduce the duration of use of the node.



At the same time, noise immunity is one of the key criteria for data transmission [8]. And it is necessary to make every effort to achieve the highest possible noise immunity, that is, reliability (to strive for 100%) [22].

Further in the article we will look at the mechanism for detecting signal collisions, and also propose our algorithm for increasing the noise immunity of a wireless network.

Related works

As the number of devices connected to the internet and the amount of data they generate increases, the wireless spectrum is becoming an essential and scarce resource [23]. In this paper [23] authors note that most connected devices use wireless technologies that use the industrial, scientific, and medical (ISM) radio bands, as a result, different technologies are interfering with each other.

In article [6] and [24] authors also write about huge number of devices expected to be connected to wireless networks. In [6] there is noted that collisions and transmission delay drawbacks are critical challenges when deploying IoT devices. And in [24] scientists pose our attention to spectrum scarcity that is one of the most important challenges currently and in the future.

Researchers [22] propose using a subset of the best radio channels (whitelisting) and to group the links per timeslot, allocating them either to the same whitelist or even appropriately reordering them to avoid collisions.

N. Sun and co-authors [25] consider underwater wireless sensor network and they note that the data transmission collision of underwater communication is very serious. Paper [26] proposes a communication protocol for low speed, long range IoT networks with a large number of IoT sensors. P. Branch and T. Cricenti in [27] write that collision management in wireless relay networks is challenging because there may be coverage overlap between multiple relays. The study [28] proposes multiple machine access learning with collision carrier avoidance protocol is proposed in the environment of 6G Internet of Things for creating an effective communication process. In [29] to solve the problem of packet collisions caused by the standard distributed coordination function protocol in wireless local area networks scientists propose a novel backoff state generation algorithm, called adaptive virtual backoff algorithm, operating in an access point. Authors in [30] propose a neighbor discovery algorithm using a bi-directional carrier sense collision avoidance and multi subchannels based on a scanbased algorithm in order to solve the problem of collisions caused by mutual interference of multiple transmitting nodes which are in one reception beam of the receiving node.

Thus, we see that the occurrence of collisions in wireless networks is a significant factor that reduces the network's noise immunity and leads to errors.

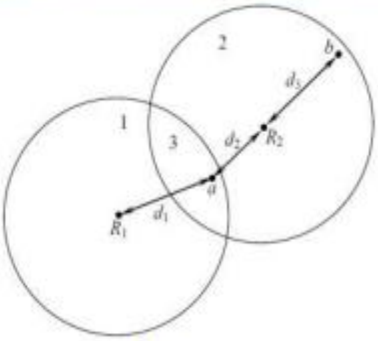
Collisions occurrence mechanism

Internet-of-Things (IoT) systems and their applications rely on sensory data for operating context-aware functions [31].

Currently, many enterprises have a variety of sensors, and these sensors record all the necessary information. It should be noted that the coverage radii of these sensors often overlap, which in turn can also lead to collisions [32]-[35].

The most significant problem of collisions manifests itself during the initial configuration of wireless networks. Its essence lies in the fact that network nodes are not synchronized from the very beginning, they do not have any information about other end nodes-neighbors and coordinator nodes that perform the functions of dividing the network into segments (subnets) of routing. Network nodes also lack information about response times. When the coordinator of the subnet requests data, all its end nodes respond almost simultaneously, which leads to the overlap of the signals transmitted by them and the destruction of information in them.

A typical example of a collision during simultaneous transmission of signals in a wireless network is shown in Figure 1. Coordinator R_1 transmits a network packet to node a, transmission signal power P. At the same time, coordinator R_2 transmits data to node b with the same signal power and in the same frequency range. The distance between R_1 and $ad1=\delta(R_1, a)>d2=\delta(R_2, a)$, so the signal from coordinator R_2 is for node a noise, the level of which will exceed the level of the useful signal from coordinator R_1, so node *a* will receive a distorted signal.



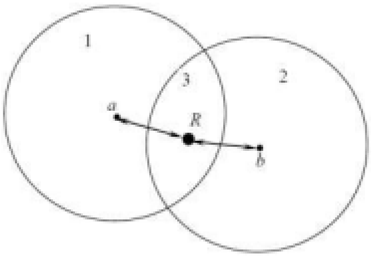
1, 2 - coordinators R_1 and R_2 signal areas; 3 – collision zone

Figure 1: An example of a collision of signals during simultaneous transmission in a wireless network

In wireless networks, the problem of collisions is complicated. The network coordinator, being in the center of the service area, ensures network interaction of several nodes that cannot directly exchange messages with each other due to insufficient signal strength and, therefore, cannot detect each other's signals. In this case, the simultaneous transmission of packets by nodes *a* and *b* to each other or to any other network node will lead to a collision in the network area served by the coordinator. An example of the impossibility of collision detection by end nodes of a wireless network in the coordinator zone is shown in Figure 2.

JOURNAL OF UNIVERSAL SCIENCE RESEARCH

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11



1, 2 – *a* and *b* nodes signals action areas; 3 – collision zone

Figure 2: An example of the impossibility of detecting a collision by end nodes a and b of a wireless network during simultaneous signal transmission

In this case, all management of collision detection and notification of end nodes is performed by the coordinator R. The collision results in the loss of transmitted data. With the described approach to the implementation of objects automated complex control system, a situation is possible when several information carriers will try to occupy the radio frequency communication channel at the same time. This will inevitably lead to the distortion of the information transmitted by them as a result of superimposing the contents of several parcels on each other. At the same time, a collision will occur. The applied coding method does not allow to separate the signals of each information carrier from the general signal, such parcels are simply not accepted. Figure 3 shows a typical example of a collision.

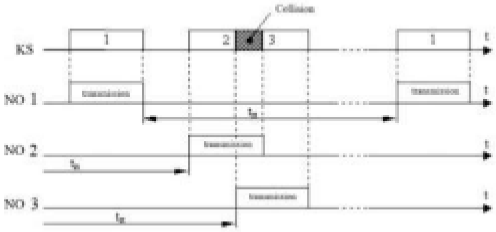
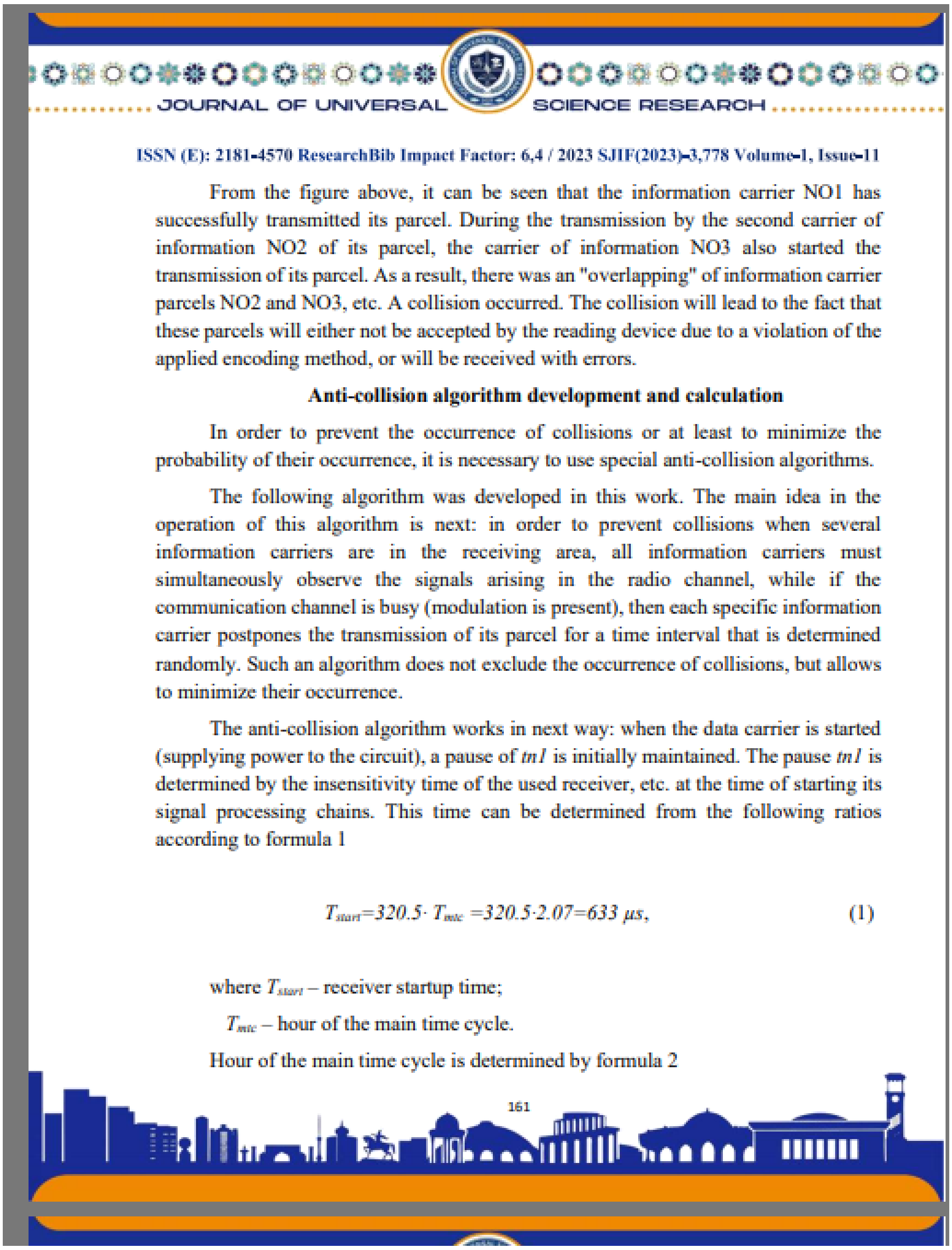


Figure 3: Collisions occurrence



JOURNAL OF UNIVERSAL SCIENCE RESEARCH

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11

From the figure above, it can be seen that the information carrier NO1 has successfully transmitted its parcel. During the transmission by the second carrier of information NO2 of its parcel, the carrier of information NO3 also started the transmission of its parcel. As a result, there was an "overlapping" of information carrier parcels NO2 and NO3, etc. A collision occurred. The collision will lead to the fact that these parcels will either not be accepted by the reading device due to a violation of the applied encoding method, or will be received with errors.

Anti-collision algorithm development and calculation

In order to prevent the occurrence of collisions or at least to minimize the probability of their occurrence, it is necessary to use special anti-collision algorithms.

The following algorithm was developed in this work. The main idea in the operation of this algorithm is next: in order to prevent collisions when several information carriers are in the receiving area, all information carriers must simultaneously observe the signals arising in the radio channel, while if the communication channel is busy (modulation is present), then each specific information carrier postpones the transmission of its parcel for a time interval that is determined randomly. Such an algorithm does not exclude the occurrence of collisions, but allows to minimize their occurrence.

The anti-collision algorithm works in next way: when the data carrier is started (supplying power to the circuit), a pause of *tп1* is initially maintained. The pause *tп1* is determined by the insensitivity time of the used receiver, etc. at the time of starting its signal processing chains. This time can be determined from the following ratios according to formula 1

$$T_{start}=320.5\cdot T_{mtc}=320.5\cdot 2.07=633\ \mu s, \qquad (1)$$

where T_{start} – receiver startup time;

T_{mtc} – hour of the main time cycle.

Hour of the main time cycle is determined by formula 2

$$T_{mtc} = (1/(f_{rf}/14)) = (1/(6.76/14)) = 2.07\ \mu s \qquad (2)$$

where f_{rf} - reference frequency, is determined by formula 3

$$f_{rf} = (f_{lg}/64) = (125/64) = 1.95\ MHz \qquad (3)$$

where f_{lg} – the frequency of the local generator, which is determined by the formula 4

$$F_{lg} = (f_r/(1+(1/125) = (125/(1+(1/125))) = 125\ MHz \qquad (4)$$

where $f_r = 125\ MHz$ - the frequency of the radio frequency communication channel.

Estimated pause time *tn1* is 5

$$tn1 = T_{start} + t_{bi} = 633 + 100 = 733\ \mu s \qquad (5)$$

where $t_{bi} = 100\ \mu s$ – hour of the bit interval, which is defined as follows 6:

$$t_{bi} = (1/V_t) = (1/10 \cdot 10^3) = 0.0001c = 100\ \mu s \qquad (6)$$

where $V_t = 10\ kBaud$ – speed of receiving data by the transmitter.

After waiting for the pause *tn1*, the receiver of the information carrier is started and the presence of the carrier frequency in the radio channel is monitored. The absence

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11

of a carrier frequency (modulation) in the radio frequency communication channel gives a signal to the microcontroller to start transmitting a parcel with information. After that, the microcontroller switches to waiting mode for the confirmation of reception/transmission and switches the receiver to active mode. If within 10 ms the receipt package was not received, the algorithm is repeated again. If the receipt/transmission confirmation parcel was successfully received, then before the next transmission of the registration parcel, a pause of *tn3* is maintained, during which the information carrier is "silent".

The duration of this pause can reach five minutes. It is necessary in order to relieve the communication channel from already registered information carriers and thereby reduce the likelihood of collisions. After the end of the *tn3* pause, the algorithm will start its work from the beginning.

If as a result of "listening" of the communication channel, the presence of modulation was detected, then the microcontroller must pause for a short random time interval before trying to transmit the package with the registration number, and then try to occupy the radio frequency communication channel again.

If the presence of modulation was detected as a result of "listening" to the communication channel, then the microcontroller must pause for a short random time interval before attempting to transmit the information packet, and then try to occupy the radio frequency communication channel again.

To do this, it generates a random integer S from the interval (0, 64). Next, a pause of *tn2*, called the delay interval, is maintained. The general random pause is selected according to the following formula 7:

$$T\sum = S \cdot tn2 \qquad (7)$$

where $T\sum$ – pause multiple of pause *tn2*, and its duration is determined by a random number S and will last until S becomes zero.

The delay interval is equal to 5 bit intervals and corresponds to the time determined from expression 8

$$tn2=(N/V_t)=(5/10 \cdot 10^3)=0.0005\ s=0.5\ ms \qquad (8)$$

where $N=5$ *bits* – number of bit intervals;

$V_t=10$ *kBaud* – speed of receiving data by the transmitter.

Thus, a random pause (without taking into account the command execution time) can take values from 0 to 32 ms.

Figure 4 shows a situation similar to that shown in (Figure 3), but when using the anti-collision algorithm.

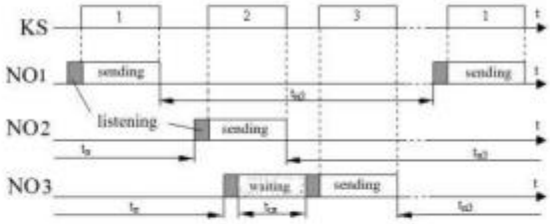


Figure 4: The method of accessing the communication channel using the anti-collision algorithm

Now, when trying to transmit the parcel, the information carrier NO 3 discovered by listening to the radio channel of the communication channel that the latter was busy and only after waiting for a random pause T_d tried to transmit its parcel again. Since the communication channel was free this time, the third carrier of information was successfully registered. Thus, this method made it possible to avoid the occurrence of a collision.

Conclusion

From the method of avoiding collisions described above, it can be seen that it is probabilistic in nature, and the probability of a carrier of information successfully

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11

obtaining a radio frequency communication channel at its disposal depends on the load of the latter. Therefore, the smaller the NO is in the range of the reading device, the less likely collisions will occur. The latter explains the need for a *tn3* pause.

The non-contact means was implemented by using a radio frequency communication channel. However, the used radio transmission system has one drawback - the occurrence of collisions when several carriers of information try to register at the same time. To solve this problem, an algorithm was developed that allows you to minimize the probability of collisions.

References:

1. Гіль, А., Чала, О., Филипенко, О. (2021). Промислові інтерфейси та протоколи передачі даних інтегрованих систем для автоматизованого управління в умовах Industry 4.0. Виробництво & Мехатронні Системи 2021: матеріали V-ої Міжнародної конференції, Харків, 127-30.

2. Чала, О., Сливка, А. (2023) Рівні засобів ІІоТ в інформаційних технологіях. Виробництво & Мехатронні Системи: матеріали VII Міжнародної конференції , Харків, С. 51-60.

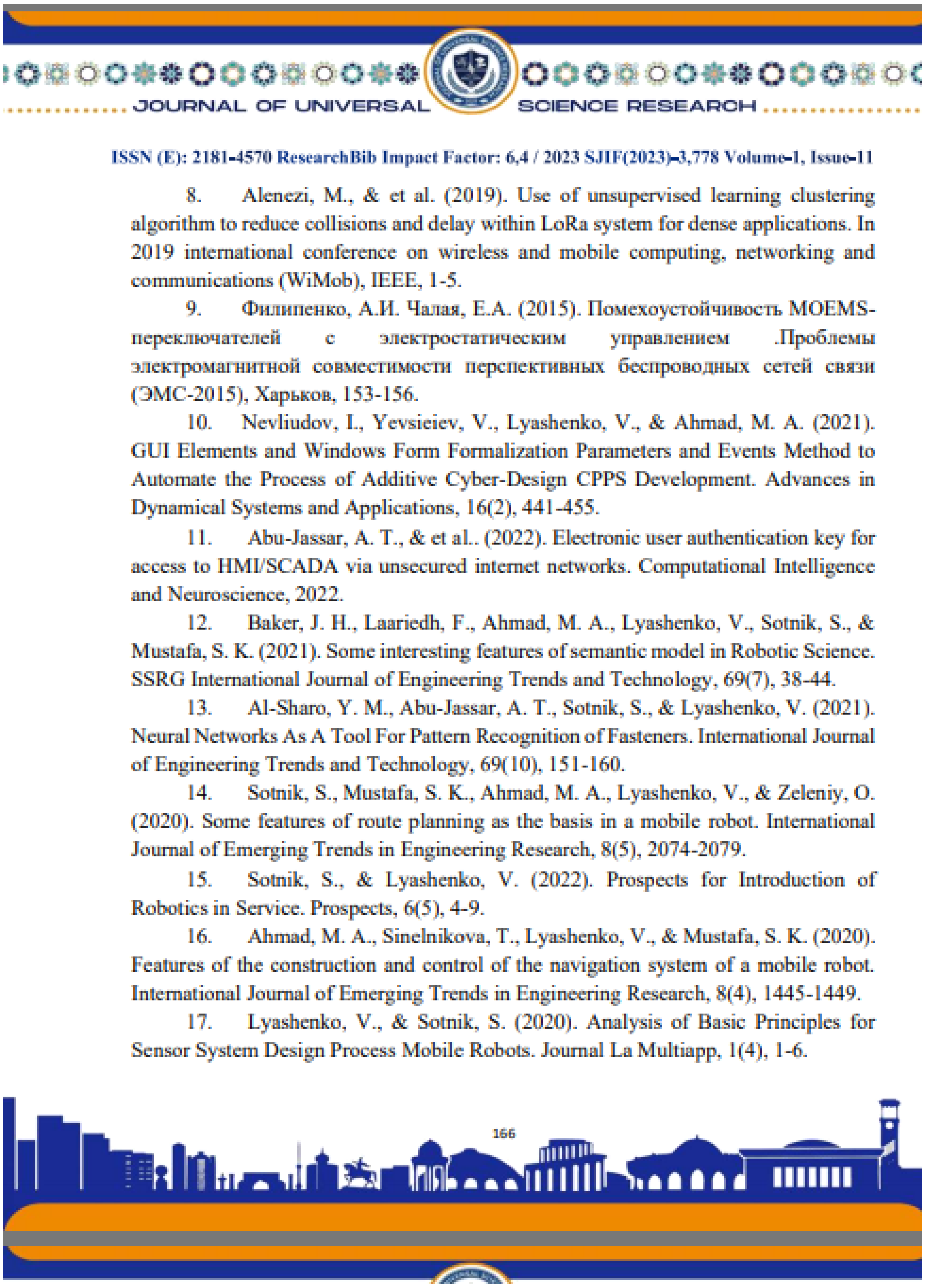
3. Lyashenko, V., & et al.. (2021). Modern CAD/CAM/CAE Systems: Brief Overview. International Journal of Engineering and Information Systems (IJEAIS), 5(11), 32-40.

4. Mustafa, S. K., & et al.. (2020). Interesting applications of mobile robotic motion by using control algorithms. International Journal of Advanced Trends in Computer Science and Engineering, 9(3), 3847-3852.

5. Nevliudov, I., Yevsieiev, V., Baker, J. H., Ahmad, M. A., & Lyashenko, V. (2020). Development of a cyber design modeling declarative Language for cyber physical production systems. J. Math. Comput. Sci., 11(1), 520-542.

6. Nevliudov, I., & et al.. (2020). Method of Algorithms for Cyber-Physical Production Systems Functioning Synthesis. International Journal of Emerging Trends in Engineering Research, 8(10), 7465-7473.

7. Невлюдов, І. Ш., Євсєєв, В. В., Андрусевич, А. О., & Максимова, С. С. (2023). Моделі та методи кіберфізичних виробничих систем в концепції Industry 4.0.



8. Alenezi, M., & et al. (2019). Use of unsupervised learning clustering algorithm to reduce collisions and delay within LoRa system for dense applications. In 2019 international conference on wireless and mobile computing, networking and communications (WiMob), IEEE, 1-5.

9. Филипенко, А.И. Чалая, Е.А. (2015). Помехоустойчивость MOEMS-переключателей с электростатическим управлением .Проблемы электромагнитной совместимости перспективных беспроводных сетей связи (ЭМС-2015), Харьков, 153-156.

10. Nevliudov, I., Yevsieiev, V., Lyashenko, V., & Ahmad, M. A. (2021). GUI Elements and Windows Form Formalization Parameters and Events Method to Automate the Process of Additive Cyber-Design CPPS Development. Advances in Dynamical Systems and Applications, 16(2), 441-455.

11. Abu-Jassar, A. T., & et al.. (2022). Electronic user authentication key for access to HMI/SCADA via unsecured internet networks. Computational Intelligence and Neuroscience, 2022.

12. Baker, J. H., Laariedh, F., Ahmad, M. A., Lyashenko, V., Sotnik, S., & Mustafa, S. K. (2021). Some interesting features of semantic model in Robotic Science. SSRG International Journal of Engineering Trends and Technology, 69(7), 38-44.

13. Al-Sharo, Y. M., Abu-Jassar, A. T., Sotnik, S., & Lyashenko, V. (2021). Neural Networks As A Tool For Pattern Recognition of Fasteners. International Journal of Engineering Trends and Technology, 69(10), 151-160.

14. Sotnik, S., Mustafa, S. K., Ahmad, M. A., Lyashenko, V., & Zeleniy, O. (2020). Some features of route planning as the basis in a mobile robot. International Journal of Emerging Trends in Engineering Research, 8(5), 2074-2079.

15. Sotnik, S., & Lyashenko, V. (2022). Prospects for Introduction of Robotics in Service. Prospects, 6(5), 4-9.

16. Ahmad, M. A., Sinelnikova, T., Lyashenko, V., & Mustafa, S. K. (2020). Features of the construction and control of the navigation system of a mobile robot. International Journal of Emerging Trends in Engineering Research, 8(4), 1445-1449.

17. Lyashenko, V., & Sotnik, S. (2020). Analysis of Basic Principles for Sensor System Design Process Mobile Robots. Journal La Multiapp, 1(4), 1-6.

18. Matarneh, R., Maksymova, S., Zeleniy, O., & Lyashenko, V. (2018). Voice Control for Flexible Medicine Robot. International Journal of ComputerTrends and Technology (IJCTT), 56(1), 1-5.

19. Mustafa, S. K., Kopot, M., Ahmad, M. A., Lyubchenko, V., & Lyashenko, V. (2020). Interesting applications of mobile robotic motion by using control algorithms. International Journal of Advanced Trends in Computer Science and Engineering, 9(3), 3847-3852.

20. Abu-Jassar, A. T., Attar, H., Lyashenko, V., Amer, A., Sotnik, S., & Solyman, A. (2023). Access control to robotic systems based on biometric: the generalized model and its practical implementation. International Journal of Intelligent Engineering and Systems, 16(5), 313-328.

21. Al-Sharo, Y. M., Abu-Jassar, A. T., Sotnik, S., & Lyashenko, V. (2023). Generalized Procedure for Determining the Collision-Free Trajectory for a Robotic Arm. Tikrit Journal of Engineering Sciences, 30(2), 142-151.

22. Kotsiou, V., & et al. (2019). Whitelisting without collisions for centralized scheduling in wireless industrial networks. IEEE internet of things journal, 6(3), 5713-5721.

23. Mennes, R., & et al. (2020). Multi-agent deep learning for multi-channel access in slotted wireless networks. IEEE Access, 8, 95032-95045.

24. Mennes, R., & et al. (2019). Deep learning-based spectrum prediction collision avoidance for hybrid wireless environments. IEEE Access, 7, 45818-45830.

25. Sun, N., & et al. (2021). Collision-free and low delay MAC protocol based on multi-level quorum system in underwater wireless sensor networks. Computer Communications, 173, 56-69.

26. Pullmann, J., & Macko, D. (2019, July). Increasing energy efficiency by minimizing collisions in long-range IoT networks. In 2019 42nd international conference on Telecommunications and Signal Processing (TSP), IEEE, 178-181.

27. Branch, P., & Cricenti, T. (2020). A LoRa based wireless relay network for actuator data. In 2020 International Conference on Information Networking (ICOIN), IEEE, 190-195.

28. Shakeel, P. M., & et al. (2021). Creating collision-free communication in IoT with 6G using multiple machine access learning collision avoidance protocol. Mobile Networks and Applications, 26, 969-980.

ISSN (E): 2181-4570 ResearchBib Impact Factor: 6,4 / 2023 SJIF(2023)-3,778 Volume-1, Issue-11

29. Kim, J. D., & et al. (2019). Adaptive centralized random access for collision free wireless local area networks. IEEE Access, 7, 37381-37393.

30. Yang, A., & et al. (2019). A bi-directional carrier sense collision avoidance neighbor discovery algorithm in directional wireless ad hoc sensor networks. Sensors, 19(9), 2120.

31. Nguyen, T. D., & et al. (2020). Fast sensory data aggregation in IoT networks: collision-resistant dynamic approach. IEEE Internet of Things Journal, 8(2), 766-777.

32. Bortnikova, V., & et al. (2019). Mathematical model of equivalent stress value dependence from displacement of RF MEMS membrane. In 2019 IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), IEEE, 83-86.

33. Nevliudov, I., & et al. (2018). Modeling MEMS membranes characteristics. In САПР у проектуванні машин. Задачі впровадження та навчання: матеріали XXVI Міжнародної українсько-польської науково-технічної конференції, Видавництво Львівської політехніки, 61-68.

34. Nevludov, I., & et al. (2017). MEMS accelerometers production technological route selection. In 2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM), IEEE, 424-427

35. Nevlyudov, I., & et al. (2016). Technological Process Identification Method for Accelerometers MEMS. CADMD 2016. XXIV International Ukrainian-Polish Conference.

ДОДАТОК Б

Код програми

Код mag_robot.py

```
from controller import Robot



robot = Robot()
timestep = 2
max_speed = 6.28

#robot.batterySensorEnable(timestep)

lm = robot.getDevice("left_motor")
rm = robot.getDevice("right_motor")

lm.setPosition(float('inf'))
rm.setPosition(float('inf'))

lm.setVelocity(0.0)
rm.setVelocity(0.0)

lidar = robot.getDevice('lidar')
lidar.enable(timestep)
lidar.enablePointCloud()

#IR = robot.getDistanceSensor("IR_1")
#IR.enable(timestep)

sensors = []
```

```
names = ["IR_1","IR_2","IR_3","IR_4","IR_5","IR_6","IR_7","IR_8"]
reading = [0,0,0,0,0,0,0,0]

previous_error = 0.0
kp = 0.6
ki = 0.0
kd = 0.1
Integral = 0.0

for i in range (0,8):
    sensors.append(robot.getDevice(names[i]))
    sensors[i].enable(timestep)

def getReading():
    for i in range (0,8):
        if int(sensors[i].getValue())>512:
            reading[i]=0
        else:
            reading[i]=1

def PID():
    error = 0
    coefficient=[-4000,-3000,-2000,-1000,1000,2000,3000,4000]
    #error=coefficient[0]*reading[0]+coefficient[1]*reading[1]....
    #...+coefficient[7]*reading[7]
    for i in range(0,8):
        error+=coefficient[i]*reading[i]
    P=kp*error
    I=Integral+(ki*error)
    D=kd*(error-previous_error)
```

```
    correction=(P+I+D)/1000
    l_speed=5+correction
    r_speed=5-correction

    if l_speed<0.0 : l_speed=0
    if l_speed>10.0 : l_speed=10.0
    if r_speed<0.0 : r_speed=0
    if r_speed>10.0 : r_speed=10.0



    lm.setVelocity(l_speed)
    rm.setVelocity(r_speed)



    #print(l_speed, r_speed, reading)

    return I, error

while robot.step(timestep) != -1:
    range_image = lidar.getRangeImage()
    #print("{}".format(range_image))
    for ri in range_image:
        if ri < 0.3:
            lm.setVelocity(0.0)
            rm.setVelocity(0.0)
            #print("Range < 0.6")
        else:
            getReading()
            Integral, previous_error=PID()
```

```
            # print ("Battery lvl:", robot.batterySensorGetValue())
```

Код maze_contr.py

```
from controller import Robot

TIMESTEP = 32
MAX_SPEED = 6.28



def  run_robot(robot):

    # екземпляри моторів

    left_motor = robot.getDevice('left_motor') # взято з моделі робота
    right_motor = robot.getDevice('right_motor')

    left_motor.setPosition(float('inf'))
    right_motor.setPosition(float('inf')) # inf = швидкість

    left_motor.setVelocity(0.0) # початкова швидкість 0
    right_motor.setVelocity(0.0)

    distance_sensor = robot.getDevice('distance_sensor')
    distance_sensor2 = robot.getDevice('distance_sensor2')
    distance_sensor.enable(TIMESTEP)
    distance_sensor2.enable(TIMESTEP)



    while robot.step(TIMESTEP) != -1:
```

```
        if distance_sensor.getValue()>0.6 and distance_sensor2.getValue()>0.6:
            left_motor.setVelocity(3.14) # початкова швидкість 0
            right_motor.setVelocity(3.14)
        elif distance_sensor.getValue()<0.6:
            left_motor.setVelocity(0) # початкова швидкість 0
            right_motor.setVelocity(3.14)
        elif distance_sensor2.getValue()<0.6:
            left_motor.setVelocity(3.14) # початкова швидкість 0
            right_motor.setVelocity(0)


        print(distance_sensor.getValue(),distance_sensor2.getValue() )


if __name__ == "__main__": # головна функція програми
    my_robot = Robot()
    run_robot(my_robot)
```

ДОДАТОК В

Демонстраційний матеріал

Позначення	Найменування	Дод. відомості
	<u>Текстові документи</u>	
1 ГЮИК. 505900.003ПЗ	Пояснювальна записка	107 арк., ф. А4
	<u>Графічні документи</u>	
2	Демонстраційний матеріал	11 арк., ф. А4

					ГЮИК. 505900.003ВД			
Змін.	Арк.	Номер докум.	Підп.	Дата				
Розроб.		Буць			Комп’ютерне моделювання сенсорної системи мобільного робота у симуляторі WeBots Відомість кваліфікаційної роботи	ЛІТ	Аркуш	Аркуші
Перевір.		Чала				Н	1	1
Т. контроль		Стародубцев						
Н. контроль						ХНУРЕ кафедра КІТАР		
Затверд.		Невлюдов						