Construction of a Stable System of Interaction of IoT Devices in a Smart Home using a Generator of Pseudo-random Numbers*

Svitlana Poperehnyak[1,*,†], Oleh Bakaiev[2,†] and Yurii Shevchuk[3,†]

[1] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute," 37 Beresteiskyi ave., 03056 Kyiv, Ukraine

[2] Institute of Software Systems of NAS of Ukraine, 40 Academician Glushkov ave., 03187 Kyiv, Ukraine

[3] DataArt, 3530 Carol Ln, Northbrook, 60062 Illinois, USA

Abstract

The article examines the areas of application of the Internet of Things, provides their classification, and considers various approaches to the categorization of IoT areas. The main attention in the work is paid to the functioning of the "Smart Home" system, its advantages and disadvantages are highlighted. In the paper, a detailed analysis of existing and promising solutions for using a pseudo-random number generator in IoT devices is carried out and the use of a pseudo-random number generator in IoT devices in a smart home is substantiated. The use of a pseudo-random number generator helps to solve the problem of resource management and simultaneous access to resources. For example, with a large number of devices in the network (lighting, cameras, sensors), randomly selected time intervals for data transmission can avoid conflicts and achieve network load balance, reducing the probability of overloading. The work presented a scheme for describing the interaction of devices, sensors, and the central control unit in a smart home with a generator of pseudo-random numbers. An improved mathematical model of the functioning of the smart home system was proposed. The model of the optimization problem proposed in the work is focused on improving the information transfer process by introducing an artificial delay, which allows to reduce the number of errors and increases the speed of data transfer. The created model takes into account the resource limitations of the pseudo-random number generator, which allows for its efficient operation with limited resources. The paper proposes the concept of building a stable system of interaction of IoT devices in a smart home, which is based on the use of a high-quality generator of pseudo-random numbers, which ensures the security of data transmission in the network and, if necessary, creates a random delay in data transmission, which can be useful for prevention of conflicts during data transmission between sensors.

Keywords

Internet of Things, IoT devices, smart home, pseudorandom number generator, devices with limited computing resources, mathematical model, multi-criteria analysis, personal data protection

1. Introduction

The Internet of Things (IoT) has rapidly evolved, enabling seamless communication between interconnected devices across various domains, including healthcare, industrial automation, smart cities, and home automation. Among these, the Smart Home concept has gained significant attention due to its potential to enhance convenience, security, and energy efficiency. However, as the number of connected devices in a smart home increases, challenges related to resource management, data transmission efficiency, and network stability become more pronounced.

One of the critical challenges in IoT-based smart home systems is simultaneous access to resources, where multiple devices—such as lighting systems, security cameras, and environmental sensors—compete for bandwidth, leading to potential data conflicts and network congestion [1]. To address this issue, the implementation of a pseudo-random number generator (PRNG) has emerged

* CPITS 2025: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2025, Kyiv, Ukraine

* Corresponding author.

† These authors contributed equally.

spopereshnyak@gmail.com (S. Popereshnyak); oleg.bakaiev@gmail.com (O. Bakaiev); shev4ukyuri@gmail.com (Y. Shevchuk)

0000-0002-0531-9809 (S. Popereshnyak); 0009-0004-5427-1196 (O. Bakaiev); 0009-0008-3331-3886 (Y. Shevchuk)

as an effective solution for optimizing communication protocols, managing resource allocation, and enhancing data security.

In addition, this study incorporates multi-criteria optimization methods to select the most effective information security measures in IoT-based smart home systems [2–4]. The optimization process considers three key criteria: effectiveness (encryption speed, reliability, and attack detection capabilities), cost (hardware expenses, energy consumption, and computational resources), and compatibility with resource-constrained IoT devices. These methods are applied to optimize encryption algorithms, authentication mechanisms, and secure communication channels while minimizing resource consumption.

2. Overview of the subject area of application

2.1. Fields of application of IoT

Let's look at the different types of fields that can benefit from the Internet of Things revolution. An outline of typical and potential fields, including health care, intelligent energy, intelligent automotive industry, industrial automation, etc., is shown in Fig. 1.

The areas of IoT were classified differently depending on the scope of functions, the number of devices required for deployment in comparison with reliability, the level of use, and other indicators [5–7], which are shown in Table 1.

Among the wide range of IoT use cases, the market is moving towards two key categories, namely mass IoT and critical IoT [6]. In mass IoT, a large number of low-cost, low-power devices typically emit a small amount of latency-free data. Devices need to send reports to the cloud regularly, so they need a smooth connection and good coverage.

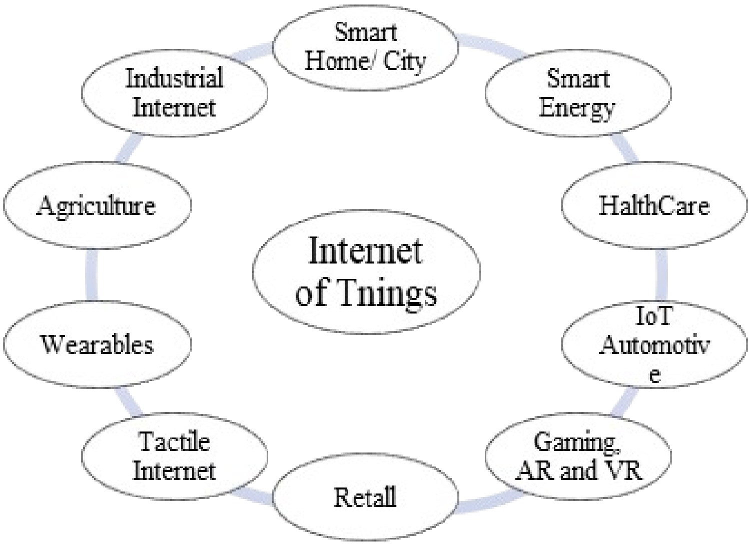

Figure 1: Classification of IoT areas

Applications of the mass Internet of Things include the smart home, intelligent agriculture, asset management, and smart metering. In contrast, mission-critical IoT applications have very high requirements for reliability, availability, and low latency.

Depending on the scope of use and adaptation of the relevant IoT sphere, they are divided into four levels of applications: infrastructure level, organizational level, individual level, and complex level. At the infrastructure level, areas such as smart cities, smart energy, smart tourism, and others are located where they, in turn, can create a new level of the ecosystem. Industrial Internet, intelligent agriculture, retail trade, and others belong to the organizational level since such programs are designed to automate the work of the organization.

Table 1
Categorization of different areas of the Internet of Things

IoT industry	Category according to [5]	Category according to [6]	Category according to [7]
Smart Home	Collaborative aware	Massive IoT	Individual level
Smart City	Collaborative aware	Massive IoT	Infrastructural level
Smart Energy	Information aggregation	Massive IoT	Infrastructural level
IoT Automotive	Information aggregation	Critical IoT	All-inclusive level
HealthCare	Information aggregation	Massive IoT	All-inclusive level
Gaming, AR, and VR	Information aggregation	Massive IoT	Individual level
Retail	Collaborative aware	Massive IoT	Organizational Level
Wearable	Information aggregation	Massive IoT	Individual level
Smart Agriculture	Collaborative aware	Massive IoT	Organizational Level
Industrial Internet	Collaborative aware	Critical IoT	Organizational Level

It is quite obvious that applications that fall under the individual level category include smart homes, games, wearable devices, and so on.

Several fields have a broader scope and can encompass all levels, including medicine and healthcare, automotive, education, and others. Table 1 presents a focused categorization of various IoT domains, while Table 2 highlights the characteristics of these well-established IoT areas.

Table 2
Characteristics of various areas of IoT

IoT industry	Data type	Feedback	Expected delay
Smart Home [8]	Stream/Historical data	Realtime	1 ms − 1000 s
Smart City [9]	Stream/Massive data	Realtime	≤ 1 ms
Smart Energy	Stream/Massive data	Realtime/ Intermittent	1 ms − 10 mins
IoT Automotive	Stream/Massive data	Realtime	≤ 1 ms
Remote surgery	Stream data	Realtime	≤ 200 ms
Remote consultancy	Stream data	Realtime	1 ms − 100 s
Gaming, AR, and VR	Stream/Massive data	Realtime	≤ 1 ms
Retail	Stream/Massive data	Realtime/ Intermittent	≤ 1 ms
Wearable	Stream data	Intermittent	Several Hours
Smart Agriculture	Historical data	Intermittent	Several Hours
Industrial Internet	Stream/Massive data	Realtime	≤ 1 ms
Tactile Internet	Stream	Realtime	≤ 1 ms

However, several fields have a wider scope and can cover all levels, such as medicine and healthcare, automotive, education, and others, see Table 1, which shows a one-sided categorization of different IoT fields, while Table 2 shows the characteristics of these already known areas of IoT.

2.2. Smart Home

A “smart home” is a modern type of residential building, organized for people to live with the help of automation and high-tech devices. A “smart” house should be understood as a system that ensures safety, comfort, and resource-saving for all users.

A smart home based on the Internet of Things uses both local (but limited) storage and processing devices (for example, a gateway or concentrator) and cloud infrastructure [10, 11]. As peripheral computing increases, performance is expected to improve significantly as operations do not require large computing resources. The gain will be in delays, load balancing, traffic reduction, and progressive resource utilization.

The “Smart Home” system has pros and cons that can play an important role in the installation of the system, let’s consider them. Similar to other devices, the Smart Home system offers several advantages that make it worth installing. These include:

- Security. The system provides comprehensive control over the premises, sending notifications in case of unauthorized access. In emergencies, the Smart Home system will attempt to prevent incidents, such as fires.
- Easy to use. The entire system is managed through a single device, typically a mobile phone.
- Flexible settings. The system allows you to tailor device settings to your preferences and modify their functions as needed. You can also add new devices to the system at any time.
- Economy. A smart home helps lower utility bills by automatically turning off devices that aren’t in use. This reduces the load on the electrical grid and decreases energy consumption. Savings can be as high as 40% on lighting and 30% on heating.
- Automation. The majority of household items can be integrated into the Smart Home system, allowing for automated control. This significantly saves time.
- Design. All system components, including buttons, thermostats, sensors, sockets, and switches, feature a modern aesthetic that seamlessly complements any interior.

The concept of “Smart home” means that housing must be equipped and designed so that all the services present, with optimal service organization, can interact with each other without great costs and complications. However, the “Smart Home” system also has its drawbacks, such as:

- Price. Although the system primarily consists of basic sensors and cameras, its price is quite high. It typically takes at least five years to recoup the investment in a Smart Home system.
- Difficulty in designing the system. A significant lack of qualified specialists in the field of electronics, programming, and design is considered a problem. Also lack high technical capabilities and experience.
- Service. Like any equipment, the system can malfunction. When this occurs, only experienced technicians can resolve the issue, and finding such professionals can be challenging. Additionally, the failure of one component can affect the operation of other connected devices.
- Limited resources. Limited resources of IoT devices mean that these devices have limited capabilities in terms of computing power, memory, and power consumption. This can significantly limit their ability to process data and interact with other devices in the IoT network.
- Security issues. This issue is identified in the following manner: an increase in the number of connected devices indicates that it can increase the risk of hacking and cyber-attacks on the system, which leads to the leakage of the owner’s confidential information and taking control of another person.
- Installation. The system is conductive, so it should be installed either immediately during the repair or before it.

3. Using a generator of pseudorandom numbers in IoT devices

3.1. Justification of the use of a generator of pseudorandom numbers in IoT devices in a Smart Home

Pseudorandom number generators (PRNGs) are essential components for ensuring the security and efficient interaction of Internet of Things (IoT) devices in a smart home. The justification for their use can be divided into several key aspects:

1. Data Security and Encryption: Encryption is required to ensure the confidentiality and integrity of data transmitted between IoT devices. PRNGs are utilized to generate cryptographic keys that secure data against unauthorized access.
2. Authentication and identity verification: For smart devices to interact reliably, they must go through an authentication process. PRNGs help generate one-time passwords, tokens, or hash values, which ensure reliable authentication of devices, preventing unauthorized access.
3. Load distribution and synchronization: A smart home consists of many devices that can work simultaneously. To effectively manage the load and prevent conflicts when accessing network resources, PRNGs can be used to distribute the time or sequence of tasks performed by devices, reducing the risk of network overload.
4. Imitation and simulation of random events: In some cases, smart homes may need to simulate random events. For example, the accidental switching on of the light or a temperature change creates the effect of the presence of people to scare off potential thieves. PRNGs provide such random changes in the operation of devices.

Thus, pseudo-random number generators play a key role in the safe, efficient, and adaptive operation of IoT devices in a smart home.

3.2. Analysis of existing and prospective solutions for the use of pseudorandom number generators in IoT devices

The main goal of the analysis of existing and promising solutions for the use of pseudorandom number generators in IoT devices is to study the effectiveness and productivity of software implementations of selected algorithms for generating pseudorandom numbers and bit sequences for building systems as part of the IoT. Recently, many review publications have been published that address the use of various pseudorandom number generation (PRNG) and bit sequence generation (PBSG) techniques in IoT systems [12]. Most of them deal with the basic aspects of using PRNG for various cryptographic purposes.

The work [13] considers various methods of obtaining random numbers, as well as the prospects of their use for IoT systems. The main emphasis is on algorithms suitable for software implementation as part of embedded software or operating systems for IoT. Special attention is paid to the cryptographic security of the proposed solutions. The article [14] provides a detailed analysis of problems in IoT systems that require the use of random number generators. For each problem, the most suitable solution is offered, in connection with which the authors analyze and evaluate in detail various methods of obtaining random and pseudo-random numbers. The performed review can be used as a recommendation and a starting point when choosing or developing a random number generator for a specific task for systems with limited resources. Some reviews are dedicated to analyzing the use of PRNG and PBSG only from the point of view of cryptography and cyber security. For instance, in [15], the authors conducted a comprehensive review of literature focused on the development, selection, and application of pseudorandom number generators in embedded systems with specific constraints, including IoT devices, wireless sensor network (WSN) nodes, and radio frequency identification (RFID) devices. The article [16] offers a review of the primary methods for generating pseudo-random numbers and their applications in cybersecurity. It analyzes the characteristics of these methods and their uses, particularly in cryptographic protection, cybersecurity, and within IoT systems. The methods for assessing the quality of source sequences and numbers are examined, along with descriptions of the primary software tools available for this purpose.

Many studies have proposed practical implementations of generators for use in embedded systems. For example, in [17], an undemanding pseudo-random number generator Arrow, which belongs to the family of Trifork generators, is proposed. This generator is based on two interconnected Lagging Fibonacci Generators (LFGs) with internal intermixing. The authors suggest that it applies to a broad range of tasks in the IoT domain. The study [18] describes two types of pseudorandom number generators: one utilizing the Bloom-Bloom-Shub (BBS) method and the other combining Xorshift with congruent generation and permutation of pseudorandom numbers. One option is proposed to be used for general-purpose purposes, while the other is for IoT devices under strictly power-constrained conditions. Hardware implementations of both methods based on FPGA programmable logic showed acceptable characteristics in terms of power consumption and performance. However, it may not be appropriate to use FPGA as the main platform in every IoT system. Also, the development of systems using programmable logic chips requires special qualifications from the developer and is quite expensive.

The article [19] describes a practical method of implementing a random number generator based on available and inexpensive components. The generator is built according to a combined scheme, where the hardware part is responsible for the formation of the entropy source based on digitized images of unpredictable signals from the environment, and the software part performs additional data processing of the entropy source. The authors do not provide data on the results of the synthesis and modeling of the proposed scheme, nor do they conduct at least a statistical evaluation of the generated sequence.

Sometimes it is suggested to use quite complex and even exotic methods to build generators. For instance, the article [20] explores the method for constructing de Braine sequence generators using shift registers with nonlinear feedback. The method proposed by the authors makes it possible to synthesize generators of pseudorandom numbers with sequences of maximum length for a given bit rate of the shift register. The scheme described in the work allows you to obtain a generator but does not provide any verification of the results of its operation or assessment of the cryptographic reliability of the proposed solution. In the article [21], a method of constructing a generator of pseudorandom numbers based on chaotic mappings is proposed. The quality of the obtained random sequence is further improved by applying the modulo-reduction function and verified by a set of mathematical and statistical tests. The encryption algorithm uses the proposed pseudo-random number generator for the secure transmission of color images received by end devices in the IoT network using the MQTT protocol over wireless communication channels and the Internet.

In this work, the method of testing pseudo-random sequences of short length based on two-dimensional statistics was used, and the criterion for testing a bit sequence of short length, which differs from the existing simultaneous use of several tests, which allows to obtain a more accurate result [22, 23].

3.3. Optimization of personal data protection in resilient IoT systems based on pseudo-random number generators

The modern development of information technologies, in particular cyber-physical systems and the Internet of Things (IoT), necessitates the improvement of methods for protecting personal data. In wartime, this problem becomes even more urgent, as cybersecurity threats increase both due to the intensification of cyberattacks and the complication of the functioning of critical infrastructures.

The use of pseudo-random number generators to optimize the protection of personal data allows the development of effective approaches to ensuring the confidentiality, integrity, and availability of personal data while optimizing the use of resources and increasing the system's resistance to external threats.

During wartime, the need to protect information from unauthorized access, interception, substitution, or analysis of traffic is especially increasing. IoT systems, which are widely used in smart homes, are vulnerable to attacks due to:

- Insufficient level of data encryption in communication channels.
- Lack of effective device authentication mechanisms.
- The ability to analyze traffic and predict device interaction.

Since a smart home is an integrated network of IoT devices, the issue of protecting the personal data of its users becomes critically important. In this context, the use of multi-criteria optimization methods allows you to find a balance between the effectiveness of protection, performance, and cost of system implementation.

One of the key elements of protecting personal data in a smart home is the cryptographic security of communications between IoT devices. However, traditional PRNGs may have certain limitations, in particular, a low level of entropy or significant consumption of resources of IoT devices. This creates the need to optimize the choice of random number generation methods, which is consistent with multi-criteria optimization approaches.

The current research question is optimizing the protection of IoT systems using a multi-criteria approach. The analysis conducted allows us to conclude that the following approaches can be used to improve the security system of a smart home:

Optimization of pseudo-random number generation algorithms:

- Analysis of PRNG resistance to cryptanalytic attacks.
- Selection of PRNG with high entropy and low resource consumption.
- Using adaptive algorithms that change generation parameters depending on the network status.
- The balance between performance and protection level.
- Choosing encryption algorithms with minimal load on IoT devices.
- Optimizing the frequency of cryptographic key updates depending on the level of threats.
- Reducing communication costs by dynamically changing the structure of the IoT device network.

Adapting the system to wartime conditions:

- Developing algorithms for backup key storage in case of system failure.
- Using fault-tolerant communication mechanisms that allow maintaining the functionality of the IoT network in difficult conditions.
- Integration of intrusion detection methods and analysis of abnormal device behavior.

Using multi-criteria optimization to build IoT networks allows:

- Optimize personal data protection algorithms, ensuring their reliability even in difficult conditions.
- Create an effective pseudo-random number generation system that provides dynamic and stable protection of IoT device communications.
- Balance the level of cryptographic protection with the resource capabilities of IoT devices, which is critically important for a smart home.
- Ensure the flexibility and adaptability of the IoT network, which will allow it to function even in conditions of external threats and changes in resource availability.

Thus, the use of multi-criteria optimization in building secure and resilient IoT networks contributes to the creation of an effective personal data protection system that meets the challenges of the modern world, particularly in wartime conditions.

4. Presentation of the main material

4.1. Definition of the problem

A smart home system utilizes various devices and sensors to offer convenient and efficient management of home environments. However, challenges can arise during its development and usage, leading to a range of potential issues.

One of the challenges faced by smart homes is **data security**. Given that these systems collect extensive information about the homeowner, including their schedule, habits, and personal details, it is essential to ensure that this data remains secure from unauthorized access. Employing a pseudorandom number generator for encryption can help safeguard this information and prevent unauthorized disclosure.

Another problem can be the **unstable operation of the system**. Since a smart home uses many different devices and sensors, it can encounter problems related to incorrect connections, equipment malfunctions, or interference between devices. Introducing an artificial delay can help reduce the load on the system and ensure more stable operation.

Resource Constraints: Many IoT devices have limited resources, such as low memory or battery power. This can lead to data retention, device reliability, and performance issues.

We will consider in detail such a drawback as the limited resources of IoT devices. Resource limitations of IoT devices mean that these devices have limited capabilities in terms of computing power, memory, and power consumption. This can significantly limit their ability to process data and interact with other devices in the IoT network.

In a smart home, where there are dozens and sometimes hundreds of different devices, limited resources can become a serious problem. For example, if sensors have limited memory, they may not store enough data for further analysis. If their computing power is limited, they may not have enough power to perform complex algorithms.

Random number generators can address this issue by enabling IoT devices to produce random numbers for various functions, such as generating encryption keys and delaying data transmission. This approach helps lower computational costs for data processing and enhances the security of data transmission. However, it’s important to consider that limited device resources may result in high computational demands for generating numerous random numbers, potentially impacting overall system performance.

Network congestion can be affected by the location of the pseudorandom number generator.

The location of the pseudorandom number generator can vary based on the specific design of the smart home system and its requirements for security and performance.

The generator of pseudo-random numbers is usually located in the central control device. It is used to generate random sequences of numbers, which are then used to generate a random data transfer delay between sensors. The sensors receive commands from the central control device to collect and transmit data, and they perform these tasks according to the specified parameters.

4.2. Algorithm of interaction between sensors, generator of pseudorandom numbers, central control device, and IoT

If the generator of pseudo-random numbers is placed in the central control device, it can be easily controlled and provide a sufficient level of security. However, this can increase the load on the central device and reduce the speed of interaction with the sensors.

If the generator is embedded within the sensors, it can alleviate the load on the central device and enable quicker interaction with the sensors. However, this approach may compromise system security, as the sensors could become more susceptible to malicious attacks.

The interaction algorithm between the sensors, the generator of pseudo-random numbers, the central control device, and IoT devices will be presented in the form of the following steps:

1. Sensors read data and transmit them to the central control device.
2. A pseudo-random number generator generates random numbers.
3. The central control device uses random numbers to generate random delays in data transmission between sensors and IoT devices.
4. IoT devices execute commands from the central control device and send reports about their status and executed commands to the central control device.
5. The central control device analyzes the reports from IoT devices and makes decisions on further management of the system.
6. The central control device sends commands to IoT devices and sets data transmission delays between sensors and IoT devices using random numbers generated by a pseudo-random number generator.

Let’s consider in more detail the components and interaction shown in Fig. 2:

- Different types of sensors (for example, motion sensors, temperature sensors, humidity sensors, smoke sensors, gas sensors, etc.) monitor the state of individual rooms or devices.
- Each sensor is connected to a local node (Node), which can be, for example, a microcontroller with a built-in Wi-Fi or Bluetooth network, or a specialized device for data collection (for example, a Zigbee or Z-Wave hub).
- Local nodes are connected to a network hub (Gateway), which can be built into the central control device or be a separate device. A gateway is used to collect data from local nodes and transfer them to the central control device.
- The central control device, which acts as the brain of the system, is a centralized place for collecting, analyzing, and processing data from various devices and sensors. This device may have a built-in pseudo-random number generator to provide random data transmission delays between sensors. Also, programs and control algorithms can be installed on the central device, which allows you to control various devices depending on the data received from the sensors.
- Users can receive information about the status of various systems and devices in a smart home through mobile devices.

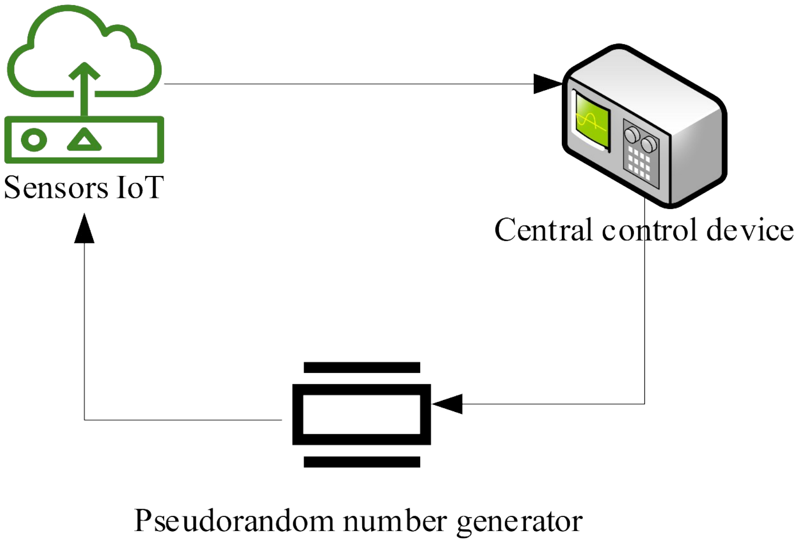

Figure 2: Scheme of interaction of IoT devices with a generator of pseudo-random numbers

The generator of pseudo-random numbers can be placed both on the sensors and on the central control device. The central control device can vary based on the specific system but is generally considered the “brain” of a smart home. It processes data from sensors and makes decisions regarding the management of home devices and systems.

In Fig. 2, each sensor transmits data to a central control device, which processes this data and makes decisions about managing IoT devices in the home. The generator of pseudo-random numbers can be located both on the sensors and on the central control device and is used in operation to ensure the randomness of the data transfer delay between the sensors.

If we consider the idea of using a generator of pseudo-random numbers proposed in this paper to ensure a random delay in data transmission between sensors, then the scheme of such interaction is as follows (Fig. 3):

1. A pseudorandom number generator (PRNG) produces random numbers.
2. The central control device generates random time intervals using PRNG and transmits them to the sensors.
3. Each sensor takes a random time interval and uses it to delay data transmission to the central control device.
4. After the delay, the sensor transmits data to the central control device.
5. The central control device processes the received data and takes appropriate actions.

In the given diagram (Fig. 3) of the interaction, a generator of pseudo-random numbers is used to generate random time intervals, which are used to delay data transmission between sensors.

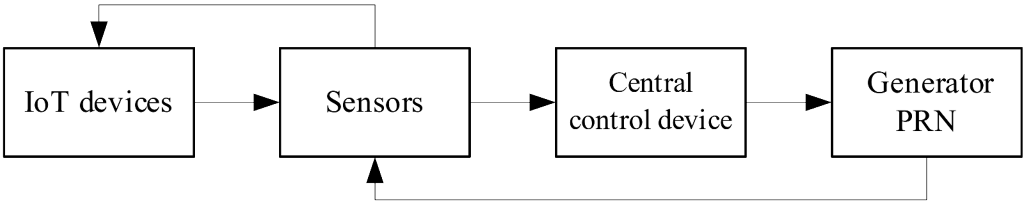

Figure 3: Scheme of interaction of IoT devices with a generator of pseudo-random numbers for the generation of random time intervals

This helps to provide a random delay in data transmission, which can be useful in preventing data conflicts between sensors operating in the Internet of Things network.

4.3. Features of the proposed model

There are different mathematical models to describe the interaction of devices, sensors, and a central control unit in a Smart Home with a generator of pseudo-random numbers. One of them is a model based on stochastic processes and information theory.

Some leading scientists in their research use the theory of stochastic processes to describe the interaction of management and control systems [24]. Another approach is to model the smart home as a multi-agent system, where each device or sensor is a separate agent with its characteristics and functionality. In such a model, agents can interact with each other and with the central control unit using special protocols and algorithms [25]. Some models utilize a system of differential equations to describe the dynamics of a smart home and its components. A special issue of the journal [26] covers recent developments in dynamical systems and differential equations, including their applications in control systems, artificial intelligence, and other areas relevant to smart homes. Prominent researchers in this field utilize systems of differential equations in their studies to model the dynamics of various devices and components within a smart home [27, 28].

However, the existing mathematical models of the interaction of devices, sensors, and the central control unit in a smart home with a generator of pseudo-random numbers are usually focused on ensuring optimal control and comfort in the home, as well as improving the quality of the interaction of devices and systems in a smart home. However, these models usually do not take into account the possibility of introducing an artificial delay to improve the information transfer process.

4.4. An improved mathematical model of the functioning of the smart home system

Let's examine an enhanced mathematical model for the operation of a smart home system, which, unlike existing models, incorporates artificial delays to optimize the information transmission process.

One of the main functions of a smart home is the interaction of sensors to collect data about the state of the premises and control their condition. To ensure the security and efficient operation of the system, pseudo-random number generators are used to generate unique sensor identifiers and random data transmission delays to avoid network conflicts.

A mathematical model of the interaction of sensors in a smart home using a pseudorandom number generator can be described as follows: suppose we have *N* sensors in a smart home, *i* each sensor sends data to the central device at a time interval of $\underline{T_i}$ (in seconds). To avoid network congestion, we want to randomly delay sending data from each sensor for a certain amount of time.

To do this, we can use a pseudo-random number generator to obtain a random delay R_i for each sensor. A pseudorandom number generator can generate numbers from the range [0,1] with a uniform distribution.

Then we can randomly delay sending data from each sensor for a time corresponding to $R_i \cdot T_i$. That is, the delay in sending data from each sensor can be calculated using the following formula:

$$Delay_i = R_i \cdot T_i, \tag{1}$$

where $Delay_i$ is a random delay for the i^{th} sensor, R_i is a random number from the range [0,1], and T_i is the time interval between sending data for the i^{th} sensor.

Therefore, each sensor sends its data to a central device with a random delay, which can help avoid network congestion and ensure an even flow of data.

The model of sensor interaction in a smart home using a pseudorandom number generator can be described as follows:

1. Each sensor has a unique identifier that is generated using a pseudo-random number.
2. Sensors periodically read data about the state of the premises and generate a random data transmission delay.
3. A pseudo-random number generator is used to ensure an even distribution of data transmission delays and avoid network collisions.
4. The data collected by the sensors are transmitted to the central system, which analyzes and monitors the condition of the premises.

Mathematically, the interaction process between sensors, devices, the central control unit, and the pseudorandom number generator can be represented by a system of equations and inequalities, where:

1. Sensors transmit data about the state of various systems of the house (temperature, humidity, movement, etc.) to the central control unit.

2. The central control unit receives data from sensors and sends control signals to devices in the house (lighting, air conditioners, heating, etc.) to maintain the set parameters in a comfortable mode for the residents of the house.
3. The generator of pseudo-random sequences is used to create random delays in the transmission of control signals to prevent resonance phenomena in the system.

Mathematically, this can be described as follows. Let sensors be denoted by *i*, sensors by *j*, and then

$$\left.\begin{matrix} x_i(t) \\ y_j(t) \end{matrix}\right\} \text{vectors of states in time t.} \tag{2}$$

Sensors transmit information to the central control unit in the form $x(t)=(x_1(t), x_2(t), \dots x_n(t))$, where *n* is the number of sensors in the house.

The central control unit receives a signal and generates a vector of control signals $u(t)=(u_1(t), u_2(t), \dots u_m(t))$, where *m* is the number of devices in the house.

Each device receives a control signal $u(t)$ and generates a state vector $y_j(t)$.

Let f_j be the processing function of the state vector $y_j(t)$ for each device. Then the general state vector $y(t)=[f_1(y_1(t)), f_2(y_2(t)), f_3(y_3(t)), \dots f_n(y_n(t))]$, where *n* is the number of devices in the house.

If the goal is to improve the process of information transmission by introducing an artificial delay, then the objective function and the system of constraints can be as follows:

Target function: Improving the quality of information transmission due to the introduction of an artificial delay. To do this, you can use the cost function, which takes into account the number of transmitted bits and the artificial delay:

$$min f(x)=w_1 \cdot tb + w_2 \cdot ad, \tag{3}$$

where tb is the number of transmitted bits; ad—artificial delay; w_1 and w_2 are weights for transmitted bits and artificial delay, respectively.

The restrictions system includes the following parameters:

- Maximum data transfer rate, which is set by the maximum data transfer rate over the network.
- Minimum artificial delay, which is set by the minimum possible delay in sending data from the sensors to the central control system.
- The maximum artificial delay is set by the maximum permissible delay of data transmission from the sensors to the central control system.

The following formulas can be used to introduce constraints to the problem:

$$\begin{aligned} tb &\leq max_{tb}, \\ ad &\geq min_{ad},, \\ ad &\leq max_{ad}. \end{aligned} \tag{4}$$

where max_{tb} is the maximum number of transmitted bits; min_{ad} is the minimum possible data transfer delay; max_{ad} is the maximum permissible data transmission delay.

The proposed optimization model belongs to the class of linear programming problems. Based on the conducted research, it was established that the optimal method of solving this problem is the use of the simplex method.

Conclusions

Building a stable system of interaction of IoT devices in a smart home based on a pseudo-random number generator provides a high level of security, adaptability, and efficiency. Data protection, effective resource management, and resistance to various cyber-attacks are achieved thanks to the use of PRNG.

The model of the optimization problem proposed in the work is focused on improving the process of information transmission by introducing an artificial delay, which allows to reduce the number of errors and increases the speed of data transmission. Also, the created model takes into account limitations on the resources of the generator of pseudorandom numbers, which allows to ensure its effective operation with limited resources.

Thus, this model is more complex and focused on a specific goal, which allows you to achieve better results in improving the process of information transfer and the effectiveness of the generator of pseudo-random numbers with a limited resource.

Declaration on Generative AI

While preparing this work, the authors used the AI programs Grammarly Pro to correct text grammar and Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] V. Sokolov, et al., Method for increasing the various sources data consistency for IoT sensors, in: IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PICST) (2023) 522–526. doi:10.1109/PICST57299.2022.10238518

[2] O. Bahatskyi, V. Bahatskyi, V. Sokolov, Smart home subsystem for calculating the quality of public utilities, in: Cybersecurity Providing in Information and Telecommunication Systems, vol. 3421, 2023, 168–173.

[3] V. Zhebka, et al., Methodology for predicting failures in a smart home based on machine learning methods, in: Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 3654, 2024, 322–332.

[4] V. Zhebka, et al., Methods for predicting failures in a smart home, in: Digital Economy Concepts and Technologies Workshop, vol. 3665, 2024, 70–78.

[5] Y. Lu, S. Papagiannidis, E. Alamanos, Internet of things: A systematic review of the business literature from the user and organisational perspectives, Technol. Forecast. Soc. Ch. 136 (2018) 285–297.

[6] B. L. R.Stojkoska, K. V. Trivodaliev, A review of internet of things for smart home: Challenges and solutions, J. Clean. Prod. 140 (2017) 1454–1464.

[7] A. A. Zaidan, B. B. Zaidan, M. Qahtan, A survey on communication components for IoT-based technologies in smart homes, Telecommun. Syst. 69 (1) (2018) 1–25.

[8] B. Zhurakovskyi, et al., Secured remote update protocol in IoT data exchange system, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, vol. 3421 (2023) 67–76.

[9] M. Moshenchenko, et. al., Optimization algorithms of smart city wireless sensor network control, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3188 (2021) 32–42.

[10] S. Adapa, Indian smart cities and cleaner production initiatives integrated framework and recommendations, J. Clean. Prod. 172 (2018) 3351–3366.

[11] A. H. Alavi, et al., Internet of things-enabled smart cities: State-of-the-art and future trends, Measurement 129 (2018) 589–606.

[12] A. Zuev, D. Karaman, Software implementation of specialized algorithms for generating pseudorandom numbers on embedded systems platforms, Control Navig. Commun. Syst. 4 (2023) 85–90.

[13] P. Kietzmann, et al., A guideline on pseudorandom number generation (PRNG) in the IoT, ACM Computing Surveys (CSUR) 54(6) (2020) 1–38.

[14] K. Seyhan, S. Akleylek, Classification of random number generator applications in IoT: A comprehensive taxonomy, J. Inf. Secur. Appl. 71 (2022) 103–365.

[15] A. B. Orue, et al., A review of cryptographically secure PRNGs in constrained devices for the IoT, in: Advances in Intelligent Systems and Computing, vol. 649, 2017, 672–682.

[16] M. A. Khomik, O. I. Harasymchuk, Application of generators of pseudo-random numbers and sequences in cyber security, methods of their construction and quality assessment, Ukrainian Sci. J. Inf. Secur. 25(3) (2023) 147–159. doi:10.18372/2410-7840.25.17940

[17] A. Orue, et al., A lightweight pseudorandom number generator for securing the Internet of Things, in: IEEE Access, vol. 5, 2017, 27800–27806.

[18] B. Paul, et al., Design and implementation of low-power high-throughput PRNGs for security applications, in: 32nd Int. Conf. on VLSI Design and 18th Int. Conf. on Embedded Systems (VLSID), 2019, 535–536.

[19] S. Popereshnyak, A. Raichev, The development and testing of lightweight pseudorandom number generators, in: 16th International Conference on Computer Sciences and Information Technologies (CSIT), 2021, 137–140.

[20] M. Miroschnyk, et al., Practical methods for de Bruijn sequences generation using non-linear feedback shift registers, in: 14th Int. Conf. on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Eng., 2018, 1157–1161.

[21] D. A. Trujillo-Toledo, et al., Real-time RGB image encryption for IoT applications using enhanced sequences from chaotic maps, Chaos, Solitons & Fractals 153(2) (2021). doi:10.1016/j.chaos.2021.111506

[22] V. Masol, S. Popereshnyak, Joint distribution of some statistics of random bit sequences, Cybern. Syst. Anal. 57(1) (2021) 139–145.

[23] V. Masol, S. Popereshnyak, Checking the randomness of bits disposition in local segments of the (0, 1)-sequence, Cybern. Syst. Anal. 56(3) (2020) 513–520.

[24] S. Gao, G. Tang, Stochastic optimal control of networked control systems with control packet dropouts, Control Theor. Technol. (2021) 2139–2156.

[25] B.-Q. Huang, Y.-L. Song, X.-C. Chen, Multi-agent Systems for smart home control: A review of Applications and challenges, Sensors 17(9) (2017) 21–35.

[26] Advances in Dynamical Systems, Differential equations, and their applications. URL: https://www.mdpi.com/journal/mathematics/special_issues/Dyn_Syst_Differ_Equ

[27] S. Fortmann-Roe, G. Bellinger, The mathematics of modeling: Differential equations and system dynamics, systems thinking & modelling. URL: https://realkm.com/2017/11/28/the-mathematics-of-modeling-differential-equations-and-system-dynamics-systems-thinking-modelling-series/

[28] Mathematical modeling of dynamic systems. URL: https://skedbooks.com/books/control-systems-1/mathematical-modeling-of-dynamic-systems/#google_vignette