Interaction of the IoT Traffic Generated by a Smart City Segment with SDN Core Network

Artem Volkov[1], Abdukodir Khakimov[1], Ammar Muthanna[1],
Ruslan Kirichek[1,2(✉)], Andrei Vladyko[1], and Andrey Koucheryavy[1]

[1] State University of Telecommunication, St. Petersburg, Russia
v.artem.nikolaevich@yandex.ru,
{abdukadir94,akouch}@mail.ru, ammarexpress@gmail.com,
{kirichek,vladyko}@sut.ru
[2] Peoples' Friendship University of Russia (RUDN University),
6 Miklukho-Maklaya St, Moscow, Russia

Abstract. The main purpose of this article is to test IoT management system based on SDN core network, as well as interaction IoT traffic with SDN-switches. To conduct investigation of management system and network infrastructure behavior we carried out several IoT traffic tests, which were generated based on partnership project oneM2M specification. In this work, we considered "Smart city" model for Central district of Saint-Petersburg (Russia). During the testing of the network infrastructure were identified several parameters such as number of simultaneously supported sessions by the Mikrotik switch using different transport protocols, was proposed a recommendation for dynamically changing virtual buffers. During the testing of the IoT management system, we define reliability of IoT data management service for determined equipment in traffic conditions of the "Smart City". Also, we estimate relationship between RTT parameter using various IoT protocols and heterogeneous traffic in conjunction with "Smart city" segment in SDN network. In order to investigate the influence of the SDN network on the RTT traffic parameter of a large number of IoT and the condition of its transmission in conjunction with heterogeneous traffic in the network, a full-scale experiment was conducted on the developed model, which in turn reflected a possible distribution scheme of a certain monitoring and control system for the Central District Saint-Petersburg. The aim of the study is also to consider the possibility of implementing the IoT data management service as a central management system for monitoring urban ecological parameters in a dense buildings environment.

Keywords: IoT · SDN · Data management · API · Smart city · Round trip time · Mikrotik switches

1 Introduction

Smart City - a futuristic idea or an inevitable future? Smart City, also known as "smart city" [1], "digital city" - a concept in which there is still no unambiguous definition. It can be stated as, smart city - a city that uses information and communication technology for

the efficient use of resources, resulting in cost savings and energy, improving the quality of life of inhabitants [2], improving the ecological environment, etc. In recent years, many cities began to implement projects related to with the use of smart city technologies. For example, Singapore (use ERP «intelligent traffic system» to deal with traffic jams.), Vancouver (Power systems organized on the principle of renewable sources), Helsinki (program to increase the availability of information resources and to attract residents to their use) and so on. Course this is only the first step toward optimizing resources and saving their possible replacement by other, less harmful to the environment, information society. However, the first step has been taken. Services such as automatic collection of indications of house energy meters [3], automation of urban transport service [4], which allows you to watch the buses moving, trolley buses and the like, and to know the approximate time of their arrival, the city multifunctional centers with electronic queue, e-health care [5], geographic information systems, and other systems that facilitate a person's life in the big city. But it's certainly not all aspects that includes the concept of "Smart City". For the full realization of these ideas, require new network infrastructure, new approaches to data organization, to meet the new requirements. As a result of these issues, there is such a thing as Big Data. It can be concluded that for effective work Smart City requires a deep integrated system consisting of many subsystems, which will take into account both the current needs of various city services, and taking into account the prospects of development in terms of new factors.

This system should be resistant to different types of impact, for example, in terms of information security; this system must be protected, so in case of fails, the work of many city services will be difficult or limited. Resistant to various force majeure, that is to be provided back-up system ready from the first time to take over the functions of failed management system "Smart City". An important aspect that should be fulfilled is the organization of such systems on the basis of generally accepted standards throughout the telecommunications world. This enables many third-party companies to provide their services on general terms and protects them from monopoly in this area.

In this paper, as one of possible services examples for "Smart City", will be considered a system for monitoring environmental parameters of air in three types of sensors, for monitoring and to develop appropriate methods to optimize the ecological situation in the dense urban and highway traffic. Just ecological picture can be useful not only in terms of the environmental situation, but also for automation of control in the field of fire and gas safety.

The approach of software-defined networks (SDN) [6] allows to significantly automate and facilitate network management by allowing them to write management applications. This approach involves the separation levels and data transmission and control, wherein the operation to determine the packet transmission route identifies a controller (the "brain" in the concept), which has information about the topology of the network status for easy monitoring and network control [7]. This functionality is determined by the controller API.

2 Specifications in the M2M and the Proposed Management System

The rapid pace of urbanization poses new challenges for city services, which certainly needs to be addressed to ensure the stable operation at such a high load. For centralized collection and management of Internet of Things, an appropriate service is required that will provide the necessary management functions. Also, for the further development of the "Smart City" architecture, required a system, which built, as already mentioned before, based on certain standards of interaction.

The need for standardization in field of machine-machine communications (M2M) has sharply increased recently in the development of infocommunications [16, 17]. To solve the set tasks, was created global partner project oneM2M [8]. The initiators of the creation were seven regional telecommunication standardization organizations ETSI, ARIB, TTC, CCSA, TIA, ATIS and TTA. Currently partnership oneM2M actively developing specifications in the field of machine to machine and the Internet of Thing, which was last updated on September 29, 2016. In total, by September 29, 2016, 24 specifications have been developed, for example TS 0001 specification - describes the functional architecture of oneM2M interaction, in TS 0003 - it displays security issues, and in TS 0009, TS 0010, implements transmission based on HTTP and MQTT protocols, respectively. Community Opendaylight, which includes a considerable number of infocommunication companies, based on the specifications of oneM2M was developed the implementation of a broker (data warehouse) with Internet of Things management functions, organized using the REST API. Internet of Things Data Management is an open source software that acts as an intermediary between oneM2M resources and related applications, with the organization having a specific application access policy to appropriate resources. All resources (Internet Things) are displayed as a hierarchical tree. Typically, the interaction with the resource tree occurs using HTTP protocols (TS 0009), CoAP (TS 0008), MQTT (TS 0010). Where TS 0008, TS 0009, TS 0010 are the corresponding oneM2M specifications. By default, applications for extracting data can work with the resource tree. But it is also possible to use control systems for devices or large data, for analytical systems. Also, are possible applications for device configuration. In this paper, as the management of "Smart City" systems to test both the services and protocols of the Internet of Things, with data organization over SDN network service IoTDM, developed Opendaylight community was chosen because of its organization in accordance with the specifications oneM2M.

The study of interaction of Internet of Things traffic with SDN [9–12] and on condition to transfer it together with heterogeneous traffic, was formed the task - to conduct SDN testing for speed the establishment of network infrastructure at the primary connecting of switches, organization of the switch buffers.

The study of the IoTDM system based on SDN was formulated the main task - to carry out testing the service in view of the real potential traffic load Internet of Things using protocols: HTTP, CoAP, MQTT, based on the model of the real area of St. Petersburg city. Testing essentially reduces to checking the interaction between IoT devices, the network infrastructure and the IoTDM service.

3 Testing Results

3.1   Testing of the Network Infrastructure

As a network infrastructure, an SDN network was built, based on Mikrotik switches supporting the openflow v.1.0 protocol and the OpenDayLight Berillium SR4 controller. As a service management acted IoTDM version Boron SR1. As data sources - developed traffic generators that implement the interaction in three protocols (HTTP, CoAP, MQTT), taking into account the specifications of the partnership project oneM2M and organization model district of St. Petersburg city.

Figure 1 and Table 1 shows the network infrastructure, organized in the SDN laboratory of the St. Petersburg State University of Telecommunications.
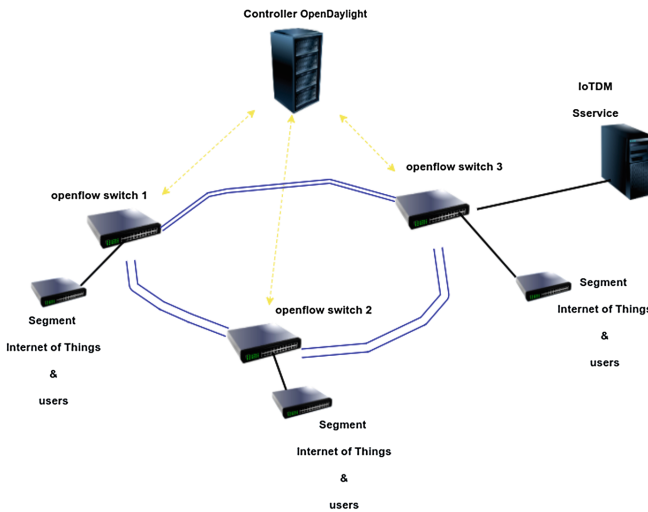
Fig. 1. The structure of the laboratory stand

As a load generator for testing the core of the SDN network, the iperf3 generator was used, which allows flexible configuration of traffic parameters. In this test, the generator was configured to simulate the load with IoT traffic [13–15]; this was reflected in the number of packets: less than 190 bytes.

In the process of work to increase the number of sessions that pass through each switch simultaneously. Testing was performed using the protocols: UDP and TCP.

The test results are shown in Figs. 2 and 3, with each of the graphs showing the dependency on each of the switches.

Table 1. The laboratory stand Structure

Hardware	Features	Functions
RAM: 16 Гб CPU: intel Xeon(R) CPU E3-1220V2	Controller	The «Brainr» for network
RAM: 12 Гб CPU: intel Xeon(R) CPU E3-1220V2	Service IoTDM	Management service for Internet of Things
Mikrotik RB201 1ui AS-RM	Network	SDN core
Cisco Catalyst 3750G Series	Network	Aggregation layer

As shown on the graphs in Figs. 2 and 3, by increasing the number of connections increases dramatically loss after a certain number of simultaneous connections in each of the SDN switches.
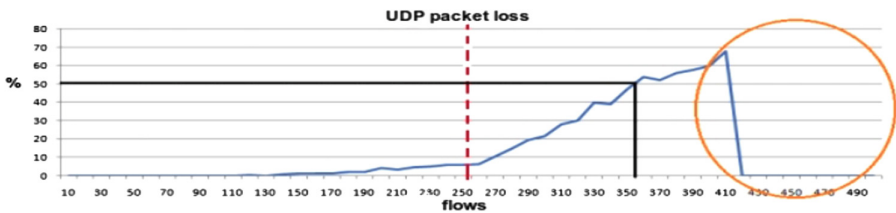
Fig. 2. Graph of UDP packet loss by increasing the number of sessions
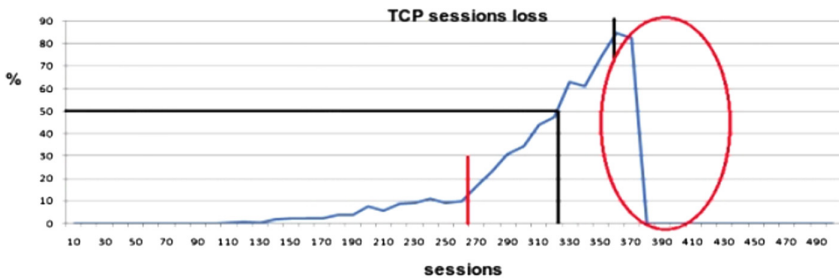
Fig. 3. Graph of the number of dangling sessions on the network by increasing their number.

Thus, it can be concluded that by using TCP, the number of possible simultaneous connections is smaller (the peak at ∼360) than when using a UDP protocol (peak at ∼400–410). A break in the graphs means the failure of the network infrastructure (shutdown of Mikrotik switches and their unplanned overload). According to the test results, we can conclude that this type of switch is possible only with the organization of internal corporate networks for small size.

Also after studying responses to switch the same type of traffic, we concluded on the use of buffers in the SDN switches. As we know, to fine-tune the buffer size in network

nodes use the formula B = P * RTT where B is the buffer size, P is the throughput, and RTT is the (Round Trip Time). With this formula, you can easily configure the optimal buffer size (queue). However, this formula does not take into account the number of parallel flows that can be different types (independent of each other).

In classical networks uses the addition to this formula $\frac{RTT*P}{\sqrt{N}}$, where N - the number of independent sessions. According to this law, determine the optimal size of the buffer. Moreover, the number N is taken as the averaged value from the exponential distribution, where P => 0.7. This method gives an error of up to 30%.

As a result of tests conducted on mikrotik equipment, we came to the conclusion that we can specify virtual buffers for each session. This will reduce the RTT parameter values for both normal network services and possibly for the tactile Internet. Since OpenFlow protocol allows to control the full resource of a network device, this task can be assigned to him, in the organization of the corresponding plug on the SDN controller.

3.2   IoTDM Service Testing as "Smart City" Service Management System

For IoTDM service testing, we created a model that highlights possible Internet of Things interaction to monitor ecological situation in the city in dense urban development. For an accurate research, we chose Saint-Petersburg central district, which has a more good shape for modelling and is divided into 6 municipal divisions: #78, Dvortsovy, Ligovka-Yamskaya, Liteyny, Smolninskoye and Vladimirsky. This allows to evenly distributing sensors when building the model. Figure 4 shows a map of the central district of St. Petersburg and its municipal divisions.
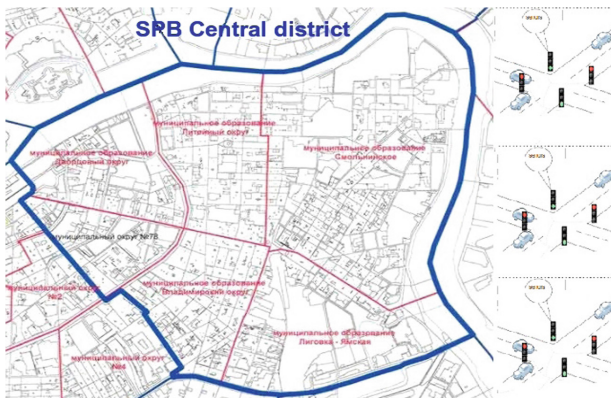
Fig. 4. Central district of St. Petersburg

We used SDN as network infrastructure, whose structure is shown in the first section.

The Smart city hierarchical model designed as follows:

- For a uniformly load distribution on network device, we connected two municipal divisions to each and every OpenFlow switch.
- In every division is involved around 40 crossroads.
- Every crossroad has four traffic lights, which serve as Internet of Thing.
- Every Internet of Thing contains three types of sensor for air ecological state control.

The model working mechanism was build based on the following algorithm:

1. Building resources tree, taking into account hierarchical division into Municipalities, crossroads, traffic lights.
2. Internet of Thing Initialization. When the IoTDM service is initialized to the resources tree, every Internet of Thing sends a request containing information about sensors readiness when connecting. Information from GPS sensor comes once.
3. Internet of Things traffic generation. Every Internet of Thing sends a request every second (Constant Instances, according to M2M specification) to the resources tree. The request contains sensors data, their registration time, value number and other metadata.

We also organized the transmission of every Internet of Thing successively from every crossroad while request to sources tree (IoTDM service) were sent simultaneously from every crossroad. Finally, we got a model with 960 Internet of Things while during one-time interval working with service of 240 Internet of Things. To build resource tree we wrote a corresponding script. To generate traffic, we developed an Internet of Thing traffic generator according to oneM2M specification and the designed "Smart city" model architecture based on HTTP, CoAP, MQTT protocols which sends requests containing, in the message body, JSON format. The IoT traffic generator was developed in Python programming language and API IoTDM.
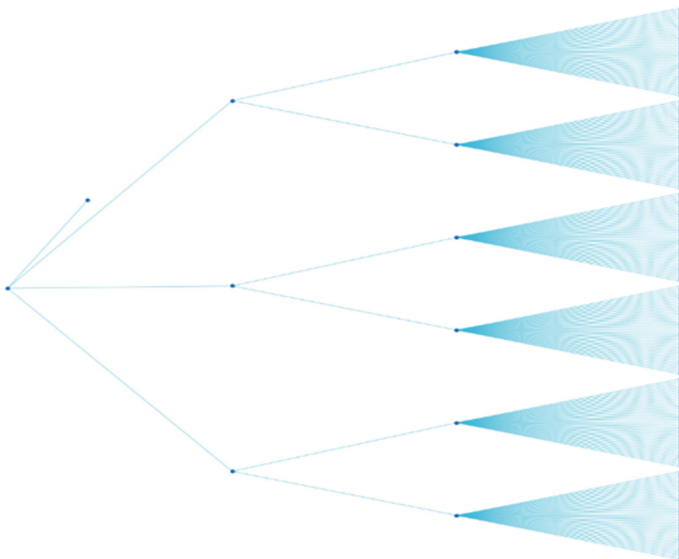
Fig. 5. IoTDM resource tree.

The built resource tree is shown in Fig. 5.

Multiple endpoints of the tree in Fig. 5, the simulated display all intersections with it at every crossroads there are 4 internet Things that are not shown in Fig. 5, with a view to properly display the tree scale. During the experiment, the RTT (Round Trip Time) parameter was calculated for each of used protocols.

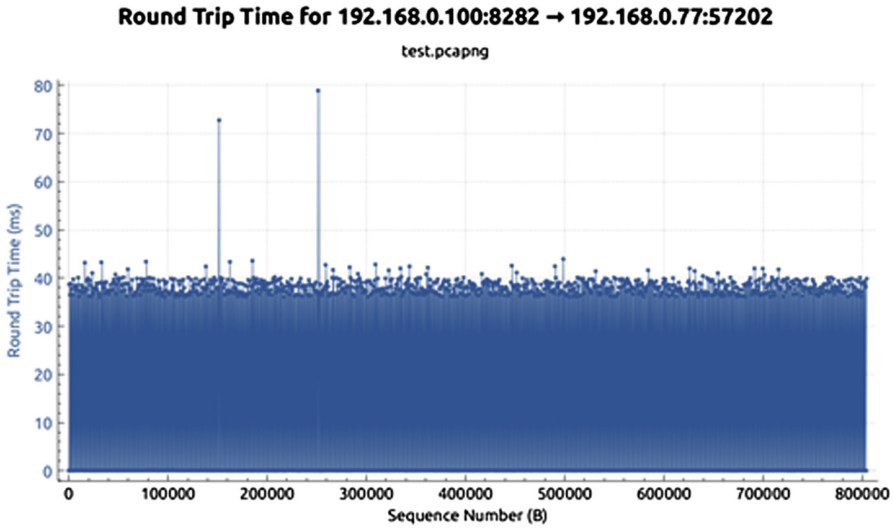The value of RTT for the HTTP protocol is displayed in Fig. 6, for CoAP in Fig. 7, for MQTT in Fig. 8.

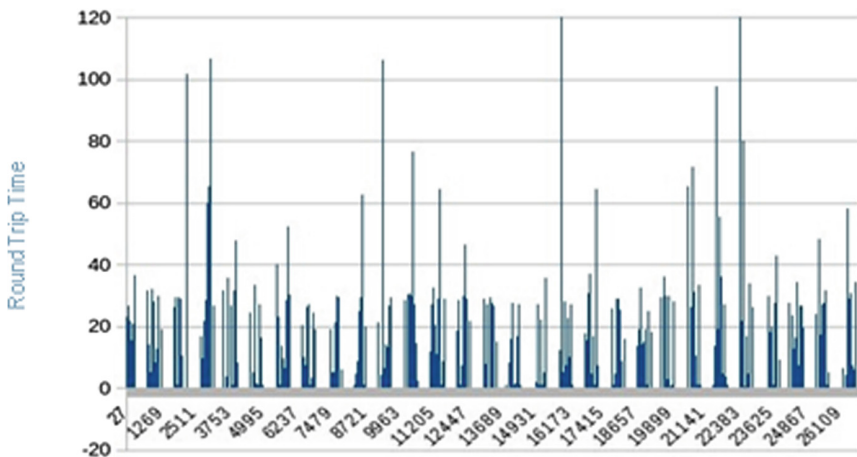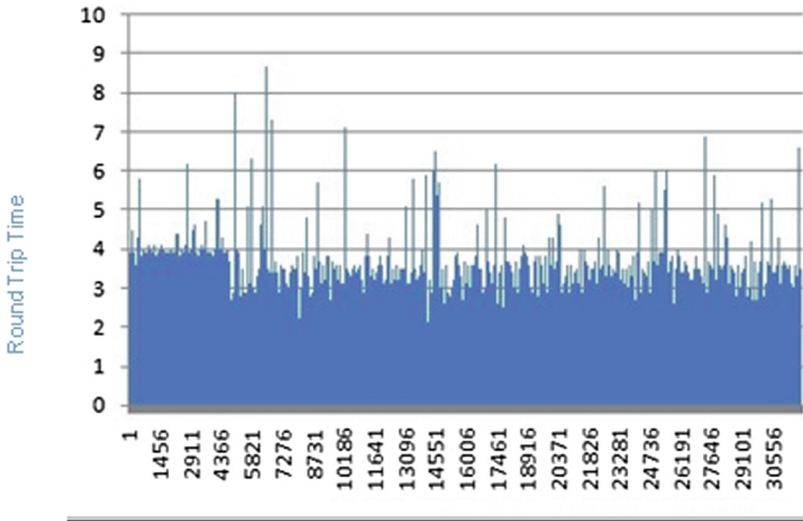**Fig. 6.** RTT for HTTP.

Fig. 7. RTT for CoAP.

Fig. 8. RTT for MQTT.

Where the number of established sessions is displayed on the abscissa, the value of Round Trip Time is displayed directly along the ordinate axis. Emissions on the graphs show the unstable operation of the service, after a certain time of testing, which consisted in closing the service ports, as a result, the delay in the successful registration of data on the IoTDM service increased. Based on the test results, we concluded which protocol is better to use for various purposes, when considering this architecture. The conclusions are shown in Table 2.

Table 2. Recommendations on the used protocols for the interaction IoT with the IoTDM service over the SDN network

Functions	HTTP	CoAP	MQTT
Internet of things registration	+	−	−
Data transmission	−	−/+	+
RTT/average value for stable operation	+($\sim$40 ms)	+($\sim$30–40 ms)	+($\sim$4 ms)

According to the data displayed in Table 2, we concluded that for the registration of Internet things to the IoTDM service, the HTTP protocol is more appropriate than others due to its work in conjunction with the transport protocol TCP, which guarantees the transmission of the registration message over the network. If, when creating the main structure or registering the Internet of Things itself, a packet will be lost on the network, this will cause a failure in the structure of the resource tree, and eventually work with the resource tree (with the entities of this branch) will be difficult. Because The IoTDM service will not be able to register the dependent entities or the data belonging to this branch and will issue an error about the unavailability of the resource.

The CoAP protocol works in conjunction with the UDP transport protocol, which eventually failed and the error described above, therefore this protocol proved to be not stable for the organization of this function. To organize the work through the protocol, the registration of the MQTT broker is required first, that is, i.e., initially work through HTTP protocol is required to organize the MQTT workflow. However, when transmitting the data itself from the Internet of Things (sensors), the MQTT protocol proved to be on the best hand, if to perform a basic comparison based on the RTT parameter. With the same load (960 IW), it was possible to achieve an RTT of 4 ms. It should also be taken into account that after prolonged operation of the generators via the HTTP protocol, the IoTDM service began to behave unstably (closing ports), which led to delays in processing the package in the service.

As a result of the Round Trip Time values, we can safely conclude that the SDN network reduces the value of the RTT parameter several times over the organization of the classical network. In the issue the testing of the IoTDM service with this load profile, the instability of its operation was revealed, namely, the ports were closed, also in result was developed generators gave an appropriate network error about the unavailability of the receiver's ports. At the same time, the further the test passed, the response time from the service increased.

We conclude that IoTDM service has a big plus that is organized in accordance with the specifications project oneM2M, but the organization of it as a management system of a service "Smart City" is required to design a distributed server architecture and thus the resource tree for the stability of its work. It also requires an accurate calculation of the hardware used in the servers for the successful processing of this amount of traffic.

4   Conclusion

The paper examined the IoTDM service as a single service for organizing a smart city management system, also in the process of experiment, taking into account that the generated IoT traffic was transmitted over an SDN network organized on the basis of Mikrotik switches, switches were tested for stable operation under heterogeneous conditions Traffic and traffic IoT smart city. As a result of the work, it was found out that the concept of building SDN networks allows to reduce such a traffic parameter as RTT by organizing the control in the form of an SDN controller, namely: the processes of creating records in the flow table and the principle of message switching (record search). As a result, the delay in switching depends only on the physical capabilities of the switch itself (ports, memory, processor speed, etc.), with the already generated flow table, most of the delay was made by the service itself, while processing requests from Internet of Things.

Recently exponential growth of Internet connected devices and network management issues have become one of the most difficult tasks. Once IoT came to life, the volume of traffic in modern communication networks is drastically changed. The lately emerged SDN approach could bring us ineluctable benefits by automating daily network engineer's chores and facilitating overall network management through its programming. In this paper were considered the advantages of SDN for management

simplification of IoT by introduction of Internet of things data management system as a service for "Smart city". The developed model, that reflected the possible distribution scheme of monitoring and management systems for the Central District of Saint-Petersburg, was used to conduct full-scale experiment.

References

1. Naphade, M., Banavar, G., Harrison, C., Paraszczak, J., Morris, R.: Smarter cities and their innovation challenges. Computer **44**(6), 32–39 (2011)
2. Bowerman, B., Braverman, J, Taylor, J., Todosow, H., von Wimmersperg, U.: The vision of a smart city. In: 2nd International Life Extension Technology Workshop, Paris (2000)
3. Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M.: Internet of Things for smart cities. IEEE Internet Things J. **1**(1), 22–32 (2014)
4. Fujdiak, R., Masek, P., Mlynek, P., Misurec, J., Muthanna, A.: Advanced optimization method for improving the urban traffic management. In: 18th Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology (FRUCT-ISPIT), pp. 48–53. IEEE (2016)
5. Kirichek, R., Pirmagomedov, R., Glushakov, R., Koucheryavy, A.: Live substance in cyberspace — biodriver system. In: 18th International Conference on Advanced Communication Technology (ICACT), pp. 274–278. IEEE (2016)
6. Alsmadi, I.M., AlAzzam, I., Akour, M.: A systematic literature review on software-defined networking. In: Alsmadi, I.M., Karabatis, G., AlEroud, A. (eds.) Information Fusion for Cyber-Security Analytics. SCI, vol. 691, pp. 333–369. Springer, Cham (2017). doi:10.1007/978-3-319-44257-0_14
7. Kirichek, R., Vladyko, A., Paramonov, A., Koucheryavy, A.: Software-defined architecture for flying ubiquitous sensor networking. In: 19th International Conference on Advanced Communication Technology (ICACT), pp. 158–162 (2017)
8. oneM2M. http://www.onem2m.org/about-onem2m/why-onem2m
9. Akyildiz, I.F., Lee, A., Wang, P., Luo, M., Chou, W.: A roadmap for traffic engineering in SDN-OpenFlow networks. Comput. Netw. **71**(9–10), 1–30 (2014)
10. Vladyko, A., Muthanna, A., Kirichek, R.: Comprehensive SDN testing based on model network. In: Galinina, O., Balandin, S., Koucheryavy, Y. (eds.) NEW2AN/ruSMART - 2016. LNCS, vol. 9870, pp. 539–549. Springer, Cham (2016). doi:10.1007/978-3-319-46301-8_45
11. Jararweh, Y., Al-Ayyoub, M., Darabseh, A., Benkhelifa, E., Vouk, M., Rindos, A.: SDIoT: a software defined based Internet of Things framework. J. Ambient Intell. Humaniz. Comput. **6**(4), 453–461 (2015)
12. Kirichek, R., Vladyko, A., Zakharov, M., Koucheryavy, A.: Model networks for Internet of Things and SDN. In: 18th International Conference on Advanced Communication Technology (ICACT), pp. 76–79. IEEE (2016)

13. Kirichek, R., Koucheryavy, A.: Internet of Things laboratory test bed. In: Zeng, Q.-A. (ed.) Wireless Communications, Networking and Applications. LNEE, vol. 348, pp. 485–494. Springer, New Delhi (2016). doi:10.1007/978-81-322-2580-5_44

14. Koucheryavy, A.: State of the art and research challenges for USN traffic flow models. In: 16th International Conference on Advanced Communication Technology (ICACT), pp. 336–340. IEEE (2014)

15. Muthanna, A., Prokopiev, A., Koucheryavy, A.: The mixed telemetry/image USN in the overload conditions. In: 16th International Conference on Advanced Communication Technology (ICACT), pp. 475–478. IEEE (2014)

16. Himayat, N., Yeh, S.-P., Panah, A.Y., Tawar, S., Gerasimenko, M., Andreev, S., Koucheryavy, Y.: Multi-radio heterogeneous networks: architectures and performance. In: Proceeding International Conference on Computing, Networking and Communication (ICNC), pp. 252–258. IEEE (2014)

17. Gerasimenko, M., Petrov, V., Galinina, O., Andreev, S., Koucheryavy, Y.: Energy and delay analysis of LTE-advanced RACH performance under MTC overload. In: Globecom Workshops (GC Wkshps), December 2012, pp. 1632–1637. IEEE (2012)