

A sensor network based on IoT and istSOS to analyse the catch basin environment in the context of the *Ae. albopictus* expansion in Switzerland.

Strigaro, Daniele (1), Cannata, Massimiliano (1), Ravasi, Damiana (2), Flacio, Eleonora (2), Antonovic, Milan (1)

(1) Institute of Earth Sciences, SUPSI, Mendrisio, Switzerland – (daniele.strigaro, massimiliano.cannata, milan.antonovic)@supsi.ch
(2) Laboratory of applied microbiology, SUPSI, Mendrisio, Switzerland – (damiana.ravasi, eleonora.flacio)@supsi.ch

KEY WORDS: istSOS, SOS, IoT, LoRa

ABSTRACT:

The continuous expansion of invasive Asian tiger mosquito, *Aedes albopictus*, combined to its ability to transmit arboviruses (e.g. dengue, chikungunya) is raising major public health concern in Europe. In Switzerland, the mosquito is firmly established in most urban areas of the Canton of Ticino, south of the Alps, and there is a real risk that it will colonize also urban areas north of the Alps in the next years. The spatial distribution and colonization of new areas by *Ae. albopictus* depends on several environmental parameters, such as winter and summer temperatures, and precipitation patterns. A key factor for *Ae. albopictus* to establish at higher latitudes is the capability to develop cold-tolerant overwintering diapausing eggs under specific environmental conditions. Weather-driven abundance models are used to map the areas of potential distribution and to predict temporal dynamics of *Ae. albopictus* and the transmission potential of arboviruses. This contribution presents the designed system that integrates low-cost and on-line IoT sensors to monitor temperature, humidity and light with istSOS an OGC Sensor Observation Service server implementation with a user friendly interface and rich feature collection to easily manage this sensor network and distribute data in a standard way (www.istsos.org).

1. INTRODUCTION

In Switzerland, as in the rest of Europe, there is an increasing awareness for the introduction and expansion of exotic invasive mosquito species. In particular, *Aedes albopictus*, also known as Asian tiger mosquito, raises concern because of its ability to transmit arboviruses (e.g. dengue, chikungunya viruses) (Flacio et al., 2015; Neteler et al., 2013; Knudsen et al., 1996). The life cycle and therefore the distribution and spread of *Ae. albopictus* is highly influenced by environmental parameters such as temperature and rain patterns (Brady et al., 2013). For example, low winter temperatures represent a key limiting factor for the survival of diapausing eggs. Indeed, Thomas and colleagues (2012) showed an increasing egg mortality when temperatures are below -10° Celsius for an extended period of time. *Ae. albopictus* can potentially reach the north of the Alps exploiting the traffic crossing the country (Müller et al., 2020). According to predictions based on a climate driven large-scale model, this mosquito species was expected to spread in part of the Swiss Plateau and in the area of the Lake Geneva while other areas, such as Zurich and Basel, seemed to have too cold winter seasons for the survival of the diapausing eggs (Neteler et al., 2013). However, the presence of urban heat islands and particular microhabitats can favour the winter survival of eggs (Ravasi et al., 2018). The monitoring of such microhabitats might allow us to better understand the environmental conditions and act in time to monitor and contain the establishment of this invasive species.

In this context, the present project aims at monitoring *Ae. albopictus* typical diapausing sites, i.e. urban storm water catch basins, and making data analysis more automated, more dynamic and efficient. These data can then be employed to build risk scenarios for *Ae. albopictus* expansion in Switzerland through the integration of multidisciplinary aspects such as mosquito surveillance expertise, data monitoring system and population modelling in order to implement a risk based

surveillance of the mosquito and optimize the costs of control treatments.

The work presented in this article on one hand tries to demonstrate how the new technologies of the Internet of Things (IoT) field, which are spreading in the market more and more, can be useful to implement custom devices at low cost to monitor adverse microhabitats with poor quality signal, such as catch basins. On the other hand, it wants to test and stress the LoRa transmission to verify if it can be a useful technology to preserve the battery and transmit real-time data.

During the first part of the project, a sensor was designed, developed and tested using the IoT technologies. Such technologies permit to build sensors with long battery life and real-time measurements transmission at low-cost, allowing the user to increase the spatial distribution of the network and the frequency of the monitoring. Once the prototype was ready to be deployed, four testing sensor networks, each composed of twenty nodes, were installed in four cities in Switzerland (i.e., Lugano, Zurich, Lausanne and Basel) to collect the following environmental data of catch basins: wall temperature, water temperature, air humidity, light and pressure. These microhabitats were selected to be monitored since they are the place where usually mosquitos deposit eggs in urban areas and can survive during the winter time. In each city, some sensors were also installed externally, inside bushes or on trees, to monitor the parameters of the surrounding area of the catch basins. The data were collected for two winter seasons (from January 2019 to February 2019 and from November 2019 to March 2020) with small technical improvements regarding the sensors from a season to the other. Data are collected thanks to the istSOS software, a Sensor Observation Service of the Open Geospatial Consortium server implementation with a user-friendly interface and useful features to easily manage the sensor network and distribute data in a standard way (www.istsos.org). In this article, the system will be described and the results of the network deployment will be presented.

2. MATERIAL AND METHODS

In the following paragraphs, we are going to describe the steps followed to implement an environmental monitoring system based on an open source approach. Starting from an overview of the system we end describing the sensor device. The aim is to implement a low-cost system that is composed by numerous nodes dispersed on a large area and that can be easily expanded with new nodes.

2.1 The system overview

The system is basically a Wireless Sensor Network (WSN) composed by nodes that sense the microhabitats and transmit data over-the-air to a data warehouse (Figure 1). The WSN covers a large area since the devices are deployed in four Swiss cities: one south of the Alps (i.e., Lugano) and the others north of the Alps but in different climatic regions (i.e., Lausanne, Basel and Zurich). As stated in the introduction, the nodes are installed inside catch basin. The nodes have sensors to measure catch basin's water and wall temperature and they guarantee a battery life of some months. During the development of the device, a particular attention is paid to the battery life, which should cover the entire winter. The winter season is important because the survival of mosquitoes' eggs is negatively affected by sub-zero temperatures. Data transmitted in real time to a data warehouse help the researchers to check the nodes' health state and get an instantaneous overview on the parameters' variations. Since an open approach is preferred and required by the project, the WSN is characterized by as much as possible open source components as well as open standards to easily share data towards other web services and increase the interoperability of the system.

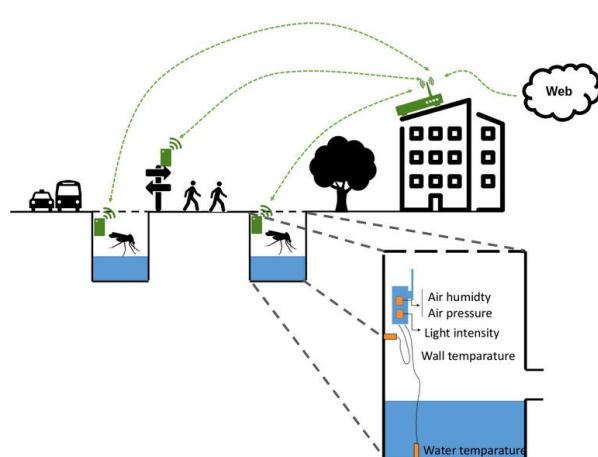


Figure 1. The system's overview.

For every group of 4-5 nodes, an equal device is positioned in some key areas where the mosquitoes usually rest (inside bushes, under trees or in other humid and protected microclimates). Data are collected every hour and then transmitted to a gateway, which re-bounces them to a server.

2.2 The data transmission

The data transmission is often one of the most power-consuming tasks of the node. Luckily, in the era of the Internet of Things (IoT), there are some options that can help in decreasing the battery consumption such as: NB-IoT, LTE

CAT-M1, LoRa (Andres-Maldonado, 2017; Hsieh, 2018; Bor, 2016). To this end, the LoRa technology is selected as media data transmission because: it is based on free radio frequency; it permits a long-range coverage; and it is the best solution to save power. In addition, LoRa combined with the LoRaWAN protocol offers critical benefits:

1. the network can be deployed with devices that do not need a sim card;
2. the network can be easily extended;
3. the transmission of the data from the device to the gateway use an end-to-end encryption;
4. costs of the data transmission are very low and potentially completely free.

During the initial phase of the project, we considered building a network with a custom gateway to handle device registration and messages coming from the nodes. This approach has the advantage to lower down to zero the costs. However, we performed some tests to understand the technology and verify the stability of the network but with poor results. Hence, we decided to adopt the Swisscom IoT LoRa network, which already covers at least 95% of the Swiss territory at an affordable cost. Such decision requires to submit the developed device to a certification process to ensure that the device respects the LoRaWAN limitations such as duty cycle, signal power, Adaptive Date Rate (ADR) and other features. This process is carried out by the ICT company Swisscom. In Figure 2, the package data transmission flow is represented. The messages sent by the devices are received by a Swisscom Gateway using an end-to-end encryption. Then, the gateway transmits the messages to a Swisscom application, which make a POST request to the SUPSI data warehouse based on istSOS.

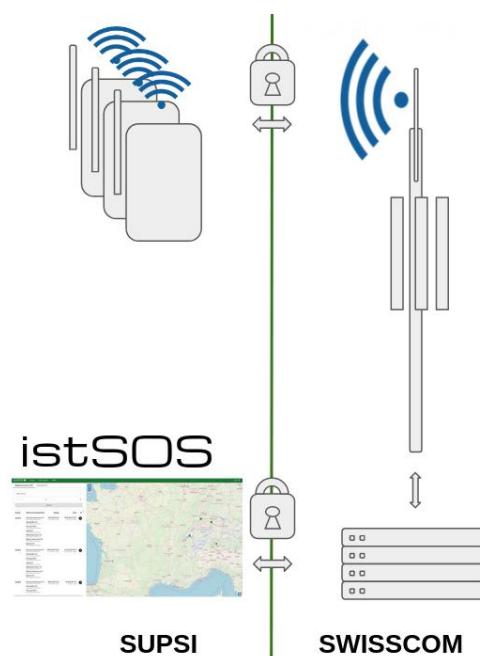


Figure 2. The LoRaWAN package transmission flow.

2.3 istSOS

The data warehouse is based on istSOS, an open source Python based implementation of the Sensor Observation Service standard (SOS) of the Open Geospatial Consortium (OGC) used in production for the management of the Hydro-meteorological

weather station network of the Canton of Ticino (Pozzoni et al. 2020). The IoT era requires a software that can manage high frequency real-time data, many spatially distributed sensors and concurrent requests performed by an always increasing number of users. Due to the analysis of the stress test performed on the software and to test the new technologies that can help in supporting high concurrent users and requests, a prototype version of istSOS (istSOS3) was developed (Cannata et al. 2019, 2018). istSOS3 permits to easily develop plugins to extend the core of the software and support other needs. Therefore, a Python plugin was developed in Python in order to add support to the POST requests coming from the Swisscom application.

2.4 The sensor node

The sensor node is one of the main objects developed during the project activity. The process that brought to the final prototype started from a very simple implementation based on wires and self-made tools to fix sensors (Figure 3). Then, after a testing phase and some fixes, a final circuit (PCB) was designed and then implemented thanks to the collaboration with the Swiss company TecInvent. The result is a battery-powered device with four sensor connections and an SD card slot mounted inside a waterproof plastic box with a transparent cover. According to the project requirements, two temperature waterproof probes based on DS18B20 are plugged to measure catch basin water temperature (°C) and catch basin wall temperature (°C). The DS18B20 probe has demonstrated very good performance in many projects (Strigaro et al. 2019) at low cost with a good stability in terms of accuracy and value drifting. In addition to catch basin temperature, other four parameters are measured: internal device temperature (°C); internal device humidity (%); air pressure (hPa); and light (lux). In Table 1, the low cost sensors selected for this project are listed. The internal temperature and humidity are useful to understand the device conditions that can be perturbed by, for example, possible water leakage. To avoid the water condensation and preserve the electronic components from corrosion, a mini pocket of silica gel was inserted in every device box. The box is an IP68 case with a transparent cover to permit the light to enter and consequently be detected by the light sensor. Instead of an expensive solar radiation sensor, the BH1750 better met the project requirements just informing if the catch basin is reached by the sunlight or not. The light of the sun can be an interesting parameter to determinate if a microhabitat is good or not for the survival of the diapausing eggs.

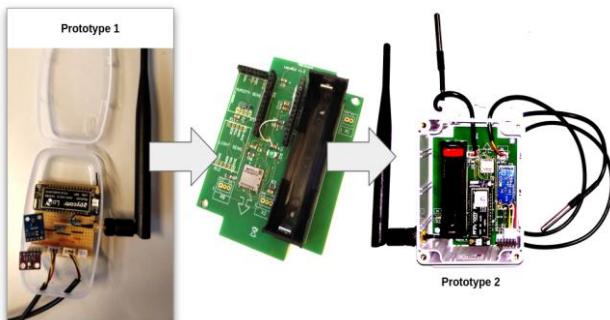


Figure 3. The prototype development.

Name	Parameters
BME280	Temperature, Pressure, Humidity
BH1750	Light
DS18B20	Temperature

Table 1. Sensors' list.

2.4.1 The LoPy microcontroller: The core of the device is the LoPy Micro-Controller Unit (MCU) from PyCom. This MCU is selected because it integrates out of the box a LoRa module compatible with the LoRaWAN. Such MCU consumes very low current (in the order of micro Ampere), during the power save/deepsleep mode. This last feature was important for our use case because the frequency of the sampling is of 30 minutes, so most of the time the sensor should just keep in memory the most significant variables, sleep and consume as low current as possible. Furthermore, the LoPy supports MicroPython, a full Python compiler and runtime that runs on the microcontroller's hardware that simplifies the code for the firmware development. At the beginning, the LoPy did not perform as expected in deepsleep mode because of an electronic bug, which was fixed with the next version named LoPy4.

2.4.2 The firmware logic: The MicroPython firmware is based on two python scripts: the first, named *boot.py*, runs once at the beginning; the second, called *main.py*, runs every time the device wakes up from sleeping (every 30 minutes). During the very first boot (or after a hardware reset), the device tries to connect to a preconfigured WiFi network to get date and time from the Internet and consequently updates these data in the internal Real Time Clock (RTC). Afterwards, the *main* cycle starts and reads the environmental data from the detected and configured sensors. Once those data are collected, the device either tries to join and pair with the nearest gateway (if it is not yet registered on the LoRaWAN network) or it restores the previous configuration obtained from the last pairing phase, which is stored into the internal flash memory. In this way, it skips the joining process and directly sends the data, saving power and reducing costs to the minimum, since every new joining process corresponds respectively to a current peak and to an exchange of messages with the gateway. The data sent are then saved to an SD card in order to keep all the data measured locally. Hence, if something goes wrong during the communication the data can be easily recovered in post-processing.

3. RESULTS

One of the main results of the previously described activities is the implementation of a node costing about 100 euros (Table 2) and a server instance of the new istSOS with the developed plugin. Hence, it was possible to assemble 100 devices. This number includes some duplicates and backup devices just in case some damages or other problems occur during the installation or the period of field work.

As already stated, during the project we deployed 75 devices in four Swiss cities (Lugano, Lausanne, Basel and Zurich) to monitor the catch-basin microhabitat during two winter seasons (2018-2019 and 2019-2020). The first device installation covered only the period from January 2019, while the second from November 2019. In Figure 4, the locations of all the sensors installed are visualized. Generally, for each city four main areas were selected based on where the oviposition traps were positioned for the routine surveillance campaign. For each

area, four devices were installed inside the catch basins and one outside.

Component	Quantity	Price in euros
LoPy4	1	30
Antenna LoRa	1	6.2
PCB Board	1	28
LiPo Battery 4000 mAh	1	3
BME280	1	4.4
BH1750	1	1.5
DS18B20	2	10.6
2GB SD Card	1	3.5
Box	1	13
Total in euros 100.2		

Table 2. Approximate costs of a single node/device.

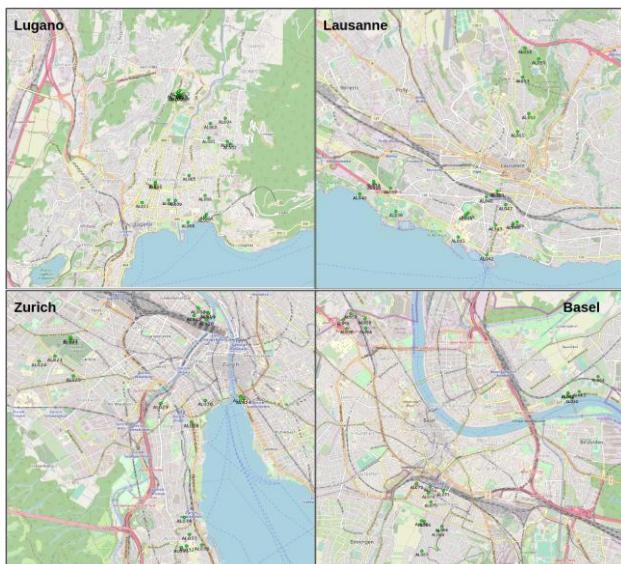


Figure 4. Maps of the 75 sensors deployed in four Swiss cities.

In Table 3 and Table 4, all the devices deployed are listed, together with the date of deployment and their life duration and. The first deployment phase started on 8th January 2019 and ended on 21st February 2019 with the installation of the nodes in Basel (Table 3). Without considering the device AL051, which had some problems from the beginning, the minimum life duration was of about 2 days for the AL058 device and 7 days for the AL050 device. The average lifetime of the sensors was of about 141 days. This means that a sensor with the right location can easily collect the data for more or less the entire winter season. The discrepancy between the lifetime of the devices is due to the quality of the signal. If the signal is poor, more power is needed to reach the gateway and so the battery life drains faster. Considering that the devices are installed inside the catch basin, such work demonstrates that the LoRa network can be used in adverse environments. The maximum lifetime value registered is 325 days for the AL054 sensor. This performance is related to the fact that the device is not inside the catch basin and consequently has a better quality of signal. The sensors nearby located in the catch basin worked for a period of around 180–200 days.

The same analysis was made for the data collected during the second winter season (Table 4). In order to increase the work time, for the second network implementation, the battery was duplicated and small improvements in the code were performed

in order to further optimize the power management of the device. With these improvements, the average lifetime of the sensors increased to 144 days. We did not notice a sensible increase of the performance, probably due to a battery module management that was built to handle only one cell and not two cells in parallel.

Device	First message	Duration
AL001	11.01.2019	72 days, 4:50:34
AL002	11.01.2019	198 days, 3:30:39
AL003	11.01.2019	123 days, 18:05:41
AL004	11.12.2019	159 days, 12:54:34
AL005	11.01.2019	191 days, 9:40:58
AL006	08.01.2019	39 days, 9:23:58
AL007	11.01.2019	210 days, 5:03:20
AL008	11.01.2019	249 days, 17:26:09
AL010	11.01.2019	70 days, 2:13:47
AL011	11.01.2019	226 days, 12:20:31
AL012	11.01.2019	159 days, 23:05:44
AL013	11.01.2019	263 days, 16:09:53
AL014	11.01.2019	248 days, 19:39:13
AL015	11.01.2019	259 days, 9:22:56
AL016	31.01.2019	204 days, 17:12:21
AL018	30.01.2019	158 days, 12:19:57
AL019	31.01.2019	108 days, 16:59:01
AL020	31.01.2019	36 days 15:47:47
AL021	01.02.2019	38 days 18:16:45
AL022	01.02.2019	79 days, 1:52:03
AL023	01.02.2019	200 days, 22:57:40
AL024	01.02.2019	20 days 15:00:31
AL025	01.02.2019	218 days, 6:24:17
AL026	30.01.2019	195 days, 22:16:32
AL027	31.01.2019	214 days, 9:56:38
AL028	31.01.2019	71 days, 23:43:12
AL029	31.01.2019	25 days, 8:11:28
AL030	27.11.2019	203 days, 14:48:56
AL031	01.02.2019	187 days, 1:45:25
AL032	01.02.2019	120 days, 12:01:57
AL033	01.02.2019	107 days, 2:13:33
AL034	01.02.2019	207 days, 16:02:04
AL035	01.02.2019	124 days, 13:18:15
AL036	12.11.2019	12 days 06:14:44
AL037	19.02.2019	200 days, 4:15:02
AL038	12.11.2019	55 days 10:57:27
AL039	19.02.2019	219 days, 11:31:18
AL040	19.02.2019	31 days 05:33:49
AL041	19.02.2019	219 days, 3:30:49
AL042	12.11.2019	176 days, 18:16:35
AL043	19.02.2019	121 days, 8:20:06
AL044	19.02.2019	209 days, 21:28:48
AL045	19.02.2019	215 days, 18:27:37
AL046	20.02.2019	187 days, 4:51:53
AL048	20.02.2019	104 days, 11:48:21
AL049	20.02.2019	217 days, 18:53:59
AL050	20.02.2019	7 days, 3:58:50
AL051	06.02.2019	0 days 01:00:00
AL052	12.11.2019	42 days 03:52:11
AL053	20.02.2019	223 days, 10:19:12
AL054	18.11.2019	325 days, 17:13:06
AL055	18.11.2019	185 days, 15:28:32
AL056	21.02.2019	206 days, 8:17:10
AL057	06.02.2019	99 days, 5:12:18
AL058	06.02.2019	2 days, 22:00:00

AL059	21.02.2019	87 days, 18:45:21
AL060	21.02.2019	223 days, 20:41:10
AL061	21.02.2019	117 days, 20:43:58
AL062	21.02.2019	158 days, 17:57:57
AL063	21.02.2019	75 days, 0:00:25
AL064	21.02.2019	146 days, 11:57:00
AL065	21.02.2019	37 days 12:28:43
AL066	21.02.2019	93 days, 10:33:57
AL067	21.02.2019	177 days, 9:02:21
AL068	21.02.2019	33 days 19:39:26
AL069	21.02.2019	46 days, 12:31:12
AL070	21.02.2019	218 days, 18:23:50
AL071	21.02.2019	164 days, 13:01:50
AL072	21.02.2019	94 days, 4:54:47
AL073	21.02.2019	86 days, 17:43:42
AL074	21.02.2019	111 days, 22:16:37
AL075	21.02.2019	136 days, 15:03:48

Table 3. Devices duration analysis of the winter season 2018-2019.

Device	First message	Duration
AL001	12.12.2019	162 days, 20:42:34
AL002	11.12.2019	190 days, 0:27:42
AL003	12.12.2019	156 days, 5:25:40
AL005	20.11.2019	15:58:07
AL007	20.11.2019	173 days, 20:44:47
AL008	12.12.2019	297 days, 11:14:19
AL010	20.11.2019	00:00:00
AL011	11.12.2019	229 days, 20:38:27
AL012	11.12.2019	72 days, 14:54:58
AL013	21.11.2019	242 days, 19:15:11
AL014	11.12.2019	237 days, 18:25:16
AL015	20.11.2019	261 days, 2:57:05
AL016	27.11.2019	209 days, 20:07:19
AL018	12.11.2019	39 days, 23:29:07
AL019	27.11.2019	152 days, 18:51:10
AL020	27.11.2019	26 days 20:10:50
AL021	27.11.2019	77 days 09:38:18
AL022	27.11.2019	171 days, 10:48:38
AL023	27.11.2019	194 days, 13:57:58
AL024	27.11.2019	16 days 17:34:30
AL025	27.11.2019	317 days, 1:37:19
AL026	12.11.2019	270 days, 11:21:01
AL027	27.11.2019	210 days, 9:42:05
AL028	20.11.2019	00:00:00
AL029	27.11.2019	183 days, 11:04:34
AL031	27.11.2019	349 days, 19:36:04
AL032	27.11.2019	239 days, 1:23:50
AL034	27.11.2019	379 days, 22:57:10
AL035	27.11.2019	61 days, 8:14:38
AL037	18.11.2019	182 days, 3:19:54
AL039	18.11.2019	11 days, 1:16:37
AL043	12.11.2019	102 days, 17:49:11
AL044	12.11.2019	274 days, 15:45:20
AL045	18.11.2019	11 days, 7:28:52
AL046	12.11.2019	150 days, 18:09:35
AL048	12.11.2019	12 days, 12:17:15
AL049	18.11.2019	35 days, 21:32:11
AL050	06.11.2019	00:00:00
AL051	06.11.2019	0 days 00:00:00
AL053	18.11.2019	55 days, 3:58:50
AL056	18.11.2019	00:00:00
AL057	13.11.2019	97 days, 16:22:18

AL058	13.11.2019	203 days, 21:19:38
AL059	28.11.2019	318 days, 12:57:33
AL060	28.11.2019	223 days, 13:32:35
AL061	28.11.2019	75 days, 19:33:48
AL062	28.11.2019	260 days, 22:06:06
AL064	28.11.2019	57 days, 21:00:04
AL065	21.11.2019	151 days, 21:07:53
AL066	28.11.2019	211 days, 6:18:31
AL067	28.11.2019	203 days, 17:29:19
AL068	28.11.2019	48 days 15:16:00
AL069	13.11.2019	00:00:00
AL070	28.11.2019	210 days, 5:23:22
AL071	28.11.2019	124 days, 14:57:07
AL072	28.11.2019	92 days, 16:34:22
AL073	28.11.2019	181 days, 2:25:19
AL075	28.11.2019	156 days, 3:20:04

Table 4. Devices duration analysis of the winter season 2019-2020.

4. CONCLUSION

The proposed study presents the results of the implementation of a WSN using an open source approach to monitor catch basin microhabitat in four cities in Switzerland. This system is built to help understanding whether the invasive mosquito *Ae. albopictus* can survive and spread in Swiss cities in the north of the Alps. In fact, even if in the north of Switzerland winters are colder than in the south, where the mosquito is already well established, the presence of such microhabitats can offer adequate environmental conditions for the surviving of diapausing eggs. Considering the final purpose of the project, the presented work tries to understand on one hand if the new IoT technologies, together with an open source approach, can help in building a WSN at low-cost and on the other if it can perform well also in adverse microhabitat with poor quality signal. To this end, IoT technologies permit to build sensors with long-range data transmission and with long battery life. Therefore, a custom device is initially built and only after a period of test a PCB is developed to make the connections more stable and easily plug together all the hardware components. The result of this activity is a sensor that can measure catch basin wall and water temperature using two waterproof probes. In addition, the device provides data about internal temperature and humidity to monitor the state inside the box. Finally, the device also measures the light in terms of lux. This last parameter can give information about the exposure of the catch basin to the light of the sun. Thanks to the deployment of 75 devices in Switzerland, it was possible to analyse the data collected and understand if the device has enough battery life to collect the data over the winter season. The results showed that such device has a mean battery life of about 140 days with a sampling frequency of one hour. This happens when the device is installed in an adverse environment, otherwise it can reach more than 300 days if it is deployed on the terrain surface where it detects a good quality of the signal. In fact, the transmission is the most consuming activity that needs to be managed in order to save power. When there is a good signal quality, the device is less eager of power.

The next steps concern a deeper data analysis to understand:

- the distribution of the quality of the signal for each of the cities considered;
- the quality of the data through the implementation of a Quality Assurance analysis based on tests (e.g range

- test, step test, time consistency test) that indicates if the data collected are meaningful or affected by biases;
- the correlation of the time series collected with the sensor installed externally and with the nearest MeteoSwiss weather station.

This last analysis helps in considering the meaningfulness of the monitoring of such harsh environments and in evaluating a future increment of the spatial distribution of the network.

In conclusion, the proposed study presents the results of the implementation of a WSN based on open and low-cost IoT technologies and istSOS to provide information related to the invasion of the *Ae. albopictus* mosquito, which can be a vector for dangerous diseases.

ACKNOWLEDGEMENTS

We thank Lukas Engeler and Valeria Guidi (SUPSI) for their precious advice on parameters collection and network deployment. We thank the municipalities of Basel, Lausanne, Lugano and Zurich for their kindness and support during the installation of the sensors. A big thank you also to Daniel Cherix, Susanne Biebinger, Gabi Müller and Barbara Wiesendanger for facilitating the inter-regional communication. We are extremely grateful to the Swiss federal office for the environment and the Swiss confederation for the financial support. This project is part of the Pilot programme 'Adaptation to climate change' of the Swiss Confederation.

REFERENCES

Andres-Maldonado, P., Ameigeiras, P., Prados-Garzon, J., Navarro-Ortiz, J., Lopez-Soler, J. M., 2017: Narrowband IoT data transmission procedures for massive machine-type communications. *IEEE Network*, 31(6), 8-15.

Bor, M., Vidler, J. E., & Roedig, U., 2016: LoRa for the Internet of Things.

Brady, O. J., Johansson, M. A., Guerra, C. A., Bhatt, S., Golding, N., Pigott, D. M., Delatte, H., Grech, M. G., Leisnham, P. T., Maciel-de-Freitas, R., Styler, L. M., Smith, D. L., Scott, T. W., Gething, P. W., Hay, S. I., 2013: Modelling adult *Aedes aegypti* and *Aedes albopictus* survival at different temperatures in laboratory and field settings. *Parasites & vectors*, 6(1), 351.

Cannata, M., Antonovic, M., Strigaro, D., Cardoso, M., 2019: Performance Testing of istSOS under High Load Scenarios. *ISPRS International Journal of Geo-Information*, 8(11), 467.

Cannata, M., Antonovic, M., Strigaro, D., Cardoso, M., 2018: istSOS Version 3. In *Free and Open Source Software for Geospatial (FOSS4G) Conference Proceedings* (Vol. 18, No. 1, p. 5).

Flacio, E., Engeler, L., Tonolla, M., Lüthy, P., Patocchi, N., 2015: Strategies of a thirteen year surveillance programme on *Aedes albopictus* (*Stegomyia albopicta*) in southern Switzerland. *Parasites & vectors*, 8(1), 1-18.

Knudsen, A. B., Romi, R., Majori, G., 1996: Occurrence and spread in Italy of *Aedes albopictus*, with implications for its introduction into other parts of Europe. *Journal of the American Mosquito Control Association*, 12(2 Pt 1), 177-183.

Hsieh, P. C., Jia, Y., Parra, D., Aithal, P., 2018: An experimental study on coverage enhancement of LTE cat-m1 for machine-type communication. In *2018 IEEE International Conference on Communications (ICC)* (pp. 1-5). IEEE.

Müller, P., Engeler, L., Vavassori, L., Suter, S., Guidi, V., Gschwind, M., Tonolla, M., Flacio, E., 2020: Surveillance of invasive *Aedes* mosquitoes along Swiss traffic axes reveals different dispersal modes for *Aedes albopictus* and *Ae. japonicus*. *PLoS Neglected Tropical Diseases*, 14(9): e0008705.

Neteler, M., Metz, M., Rocchini, D., Rizzoli, A., Flacio, E., Engeler, L., Guidi, V., Lüthy, P., Tonolla, M., 2013: Is Switzerland suitable for the invasion of *Aedes albopictus*? *PLoS One*, 8(12), e82090.

Pozzoni, M., Salvetti, A., Cannata, M., 2020: Retrospective and prospective of hydro-met monitoring system in the Canton Ticino, Switzerland. *Hydrological Sciences Journal*, 1-15.

Ravasi, D., Guidi, V., Flacio, E., Lüthy, P., Perron, K., Lüdin, S., Tonolla, M., 2018: Investigation of temperature conditions in Swiss urban and suburban microhabitats for the overwintering suitability of diapausing *Aedes albopictus* eggs. *Parasites & vectors*, 11(1), 1-9.

Strigaro, D., Cannata, M., Antonovic, M., 2019: Boosting a weather monitoring system in low income economies using open and non-conventional systems: Data quality analysis. *Sensors*, 19(5), 1185.

Thomas, S. M., Obermayr, U., Fischer, D., Kreyling, J., Beierkuhnlein, C., 2012: Low-temperature threshold for egg survival of a post-diapause and non-diapause European aedine strain, *Aedes albopictus* (Diptera: Culicidae). *Parasites & vectors*, 5(1), 1-7.