Experiments for Fire Detection Using a Wireless Sensor Network

Ronald Beaubrun
Department of Computer Science and Software Engineering
Université Laval
Quebec, Qc, Canada
e-mail: ronald.beaubrun@ift.ulaval.ca

Yacine Kraimia
Department of Computer Science and Software Engineering
Université Laval
Quebec, Qc, Canada
e-mail: yacine.kraimia.1@ulaval.ca

***Abstract*— Recent works have shown the needs for using *Wireless Sensor Networks* (WSN) for rural and forest fire detection. In this context, a number of approaches have been proposed. However, such approaches do not take the wind direction into account, which is not realistic and does not provide accurate information on how the actual fire can be spreading out. This paper proposes a set of real-time experiments that take into account the wind direction in the context of fire detection using a WSN. Results analysis shows that the wind direction has an impact on the temperatures recorded by the sensors and can be an important parameter for on-time fire detection.**

Keywords-Experiment; fire detection; wind direction; wireless sensor network (WSN).

I. INTRODUCTION

Nowadays, the necessity for observing and controlling hostile environments through *Wireless Sensor Networks* (WSNs) becomes essential for many military and scientific applications [1]. Recent works have shown the needs for using such networks for rural and forest fire detection [2]-[8]. The basic principle for detecting fire using WSNs is to deploy a number of wireless sensors to cover a target area. Such sensors should be able to collect several parameters, such as temperature, humidity level and light intensity [2], [4]. In order to properly evaluate those parameters, three types of sensors are generally used [8]. The first type is responsible for collecting information about the environment where these sensors are deployed. Such information is transmitted via a radio link to the second type of sensors, *i.e.*, the sinks. This type of sensors processes the data coming from the wireless sensors and transmits such data to a gateway. The latter is directly connected to a host computer via a USB (*Universal Serial Bus*) port, which enables to directly transfer the data received from the sinks to the host computer. In this context, the host computer displays the processed information to home users.

Many applications of WSNs have been implemented for fire detection [1], [9], [10]. The role of these applications is to provide accurate information on how the actual fire is spreading out. In this context, a number of equipments, such as GPS (*Global Positioning System*), cameras and base stations, can also be used in order to have more details and to better locate the fire. For example, Lloret *et al*. [11] used cameras in a wireless sensor network for image and video caption of a real fire, which is useful to validate the presence of fire and prevent false alarms. Also, the sensor network implemented by Osman *et al*. [12] does not take the wind into account, which does not properly detect the presence of fire on time. In the same vein, Yu *et al*. [13] propose a method for detecting forest fire, using WSNs for collecting information on temperatures, relative humidity and wind speed. However, in the paradigm proposed in [13], the wind direction is not taken into account, which is not realistic and arises some questions. How can the presence of wind in a site have an impact on the temperatures recorded? Does the wind direction affect the process of fire detection?

This paper proposes a set of real-time experiments that enable to assess the impact of wind direction on the temperatures collected by a WSN. It is organized as follows. Section 2 presents the hardware platform, as well as the application implementation of the proposed approach. Section 3 describes the experiments and discusses the results analysis. Section 4 gives some concluding remarks.

II. THE PROPOSED APPROACH

In this research, we consider a realistic model which takes into account the wind direction in the context of fire detection using a WSN. In this section, we present the hardware platform and the implementation of the WSN applications.

A. Hardware platform

The proposed network consists of a sensor that collects information about its environment, and a sensor gateway that transfers the collected data to a host computer. In order to ensure interaction between the environment and the hardware system, a primary application for data collection is implemented, whereas a second application for graphical display is installed on the host computer.

 ISBN: 978-1-61208-207-3

For the experiments, *Tmote Sky* wireless sensors from Moteiv were used [14]. Each sensor is equipped with integrated detectors for light, temperature and humidity detection [15]. The sensors are controlled by a Texas Instruments MSP 430 microcontroller which has 48 KB of flash, 10 KB of RAM, and contains an internal digitally controller directed oscillator (DCO) that runs at 8 Mhz. In the same vein, *TinyOS* was used as a development environment [16]. It is an operating system designed for networks with limited resources. TinyOS libraries and applications are written in *nesC*, a version of *C* that was designed for programming embedded systems. In nesC, programs are built from components that are connected together to form an entire program.

According to [15], the units for the temperatures obtained are ADC (ADC_{counts}). Conversion to °C is done as follows:

- Find the internal voltage V_{int} as follows :

$$V_{int} = (ADC_{counts}) * (V_{ref}/4096) \quad (1)$$

where $V_{ref} = 1.5$

- From (1), it is possible to calculate the temperature in °C as follows:

$$Temperature\ (°C) = (V_{int} – 0.986)/0.00355 \quad (2)$$

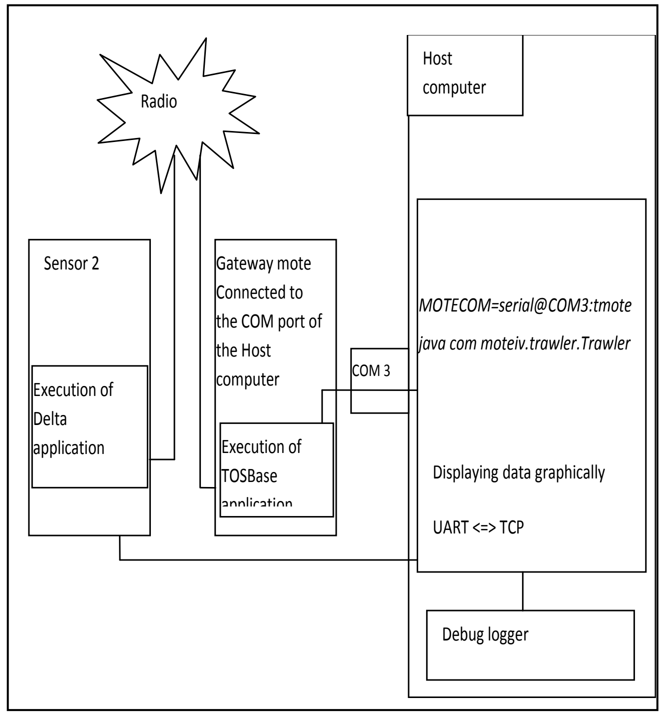

Figure 1. Deployment diagram.

B. *Implementation of the WSN applications*

A WSN application is a distributed program which consists of several modules that are executed by different computers. Such modules are illustrated in the deployment diagram presented in Figure 1, where the nodes represent the software packages and the lines correspond to the data flowing between the nodes. The application execution is controlled by a host computer that is connected with the sensor gateway. Then, the communication between the sensor gateway and the host computer is established with a USB cable, using the application coming from the base station (*apps/TOSBase part*) of TinyOS distribution.

In the context of this research, the needs for implementing a number of applications are specific. First of all, it is necessary to be able to modify the parameters of the experiments: the fire intensity, the wind direction, as well as the number of sensors and their coordinates. Secondly, as the deployment is expensive, it is necessary to place the sensors in specific locations, so that it is possible to download all the samples recorded and to detect missing data. Furthermore, it is difficult to control the wind direction. In order to do so, a controlled wind will be produced by using a fan. Finally, an electric oven will be used in order to generate and simulate the heat.

Moreover, the main application used in the experiments is called *Delta*. Its installation requires first to compile it in order to obtain a binary image in TinyOS, then to install the binary image on each sensor. The commands for compiling and installing Delta application are the following:

cd /opt/moteiv/apps/Delta
make tmote
make tmote reinstall,1

Once Delta application is installed, each sensor node can sample its internal temperature and communicate it to the other sensors. Then, another application called *Trawler* that comes with TinyOS can be used. Trawler enables to start the process of creating an ad hoc network and displaying the network topology, as well as the messages received by the PC on the screen. It is responsible for sending the sensor commands via a USB port and the gateway. Also, it is responsible for graphical representation of incoming data, and for displaying the current temperatures obtained from the sensor network and information about the sensor status. The command for running Trawler is:

MOTECOM = serial@COM4:tmote java com moteiv.trawler.Trawler

This command enables to send requests to the sensor nodes through port COM4 that connects the gateway with the PC.

Moreover, the following application *java net.tinyos.sf.SerialForwarder* is used as a java applet that creates a TCP socket for enabling data sharing with other applications on the host computer. It forwards the requested packets to the *Universal Asynchronous Receiver Transmitter* (UART), as the sensor attached to the PC sends the messages on the radio link. In addition, it takes the messages coming from the UART, and forwards them to other users via an Internet connection, which enables other applications to use such information.

Note that, in the proposed experiments, only one sensor sends information to the sensor gateway using the radio link. Then, the sensor gateway sends such information to the host computer which displays it on the screen. In this context, the sensor gateway runs an application that listens to the radio link. When a message coming from the sensors is received, it is transmitted via a serial interface, in order to make it available to the *serial forwarder*, and finally to the screen.

III. RESULTS AND ANALYSIS

The first experiment aims at collecting the temperatures recorded and transmitted over time by the sensor. Such an experiment is mainly carried out for testing the sensor capability to collect exact temperature values. For that, an oven is used to produce the heat, as the sensor is placed at 6 cm from the heat and the wind speed is set to zero. At the beginning of the experiment, the sensor indicates 17,079° C as ambient temperature. Then, after turning on the oven, the sensor node indicates that the temperatures are increasing, as illustrated in Figure 2. We then realize that the temperature variation is linear from the 103rd to the 1203rd second, then becomes logarithmic. The maximum temperature is 54.215° C obtained after 2303 seconds.

Such results can be interpreted as follows. At the beginning of the fire, the temperatures increase quickly, which justifies the linearity between 103 and 1203 seconds. After a while, the growth rate of the temperatures decreases until it converges to a nearly constant value. This corresponds to the case of a real fire, where temperatures are used to continuously increase until a fixed value [13]. In other words, in the context of fire monitoring, a 50° C threshold can be set. Therefore, when the temperatures collected by the sensors exceed such a threshold, an alarm can indicate the possible presence of fire.

Moreover, in the context of a real wildfire, the wind can play an important role in the fire propagation. The purpose of the following experiments is to evaluate the impact of the wind direction on the temperatures recorded by the sensors. It helps answer the following question: will the temperatures collected by the sensors remain the same if the tests are repeated under the same conditions, but with changes in the wind direction? For this, a wind generated by a fan at a constant speed of 8 km per hour is established. The distance between the sensor and the heat is set to 6 cm. The wind direction is controlled by positioning the fan in specific angles relative to the axis that passes through the heat and the sensor as follows: 0 degree, 45 degrees, 90 degrees, 135 degrees and 180 degrees, which is illustrated in Figure 3. More specifically, angle of 0 degree means that the wind blows in the same direction as the heat and the sensor, *i.e.*, the wind first passes through the heat before it reaches the sensor. On the other hand, angle of 180 degrees means that the wind first passes through the sensor before it reaches the heat. Angles of 45 degrees, 90 degrees and 135 degrees are other directions that the fan can take with respect to the heat and the sensor. Note that for all the experiments, the temperature values are registered at the same time (2200th second) for each orientation. Also, note that we have not considered any gust factor in the experiments.

Figure 4 illustrates the temperature values obtained for this experiment. We realize that the highest temperature is registered when the wind direction makes an angle of 0 degree. Every time the wind direction is changed, the sensor collects lower temperature values. Specifically, at 135 degrees, the lowest temperature is registered. Indeed, when the wind blows at 135 degrees, it redirects the heat from the fire to other directions. As a result, it prevents the sensors from collecting the correct temperature values. Theoretically, the sensor should collect the lowest temperature at 180 degrees. However, for unexplained reasons, it was not possible to obtain such a temperature for this condition of the experiment.

In general, the distance between the sensor nodes and the heat can play an important role in the early detection of the fire. As a result, the sensor nodes must be as close as possible to the heat, so that they can report the presence of fire on time. In the next experiment, we propose to evaluate the impact of wind direction on the temperatures collected, while changing the distance between the sensor and the heat to 20 cm, and keeping the other conditions the same as in the previous experiment. Figure 5 illustrates the results obtained for this experiment. As for the previous experiment, the highest temperature is registered for wind direction of 0 degree, whereas the lowest temperature is registered for wind direction of 135 degrees.

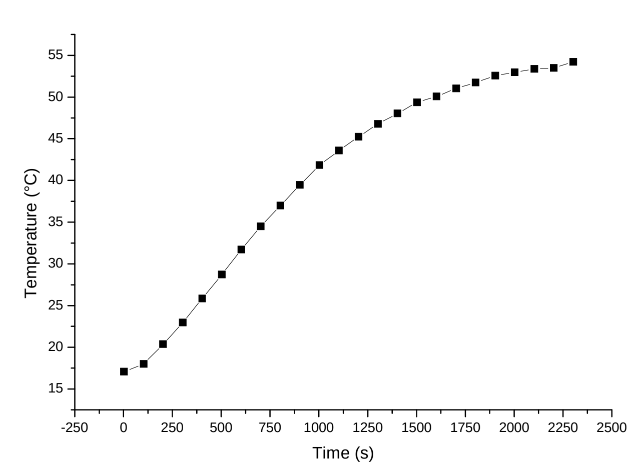

Figure 2. Temperature variation indicated by the sensor node.

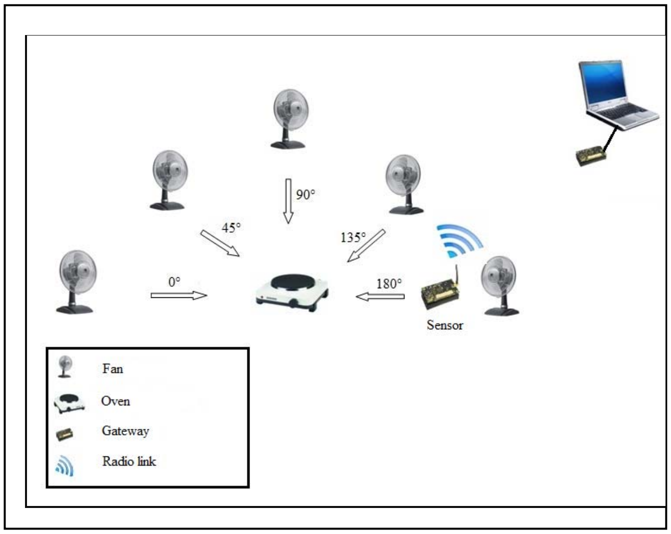

Figure 3. Variation of wind direction in the proposed approach.

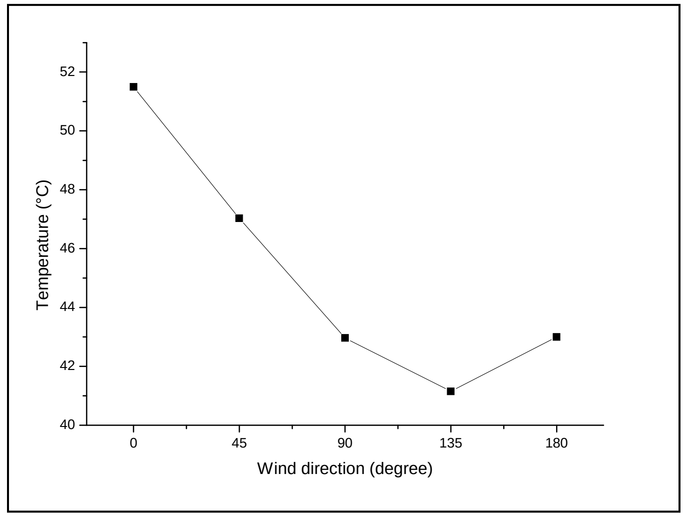

Figure 4. Temperature variation in function of the wind direction at 6 cm.

Figure 6 compares the results obtained from both experiments. In general, for the same wind direction, the temperatures collected by the sensor at 20 cm from the heat are lower than those collected at 6 cm. For instance, the highest and lowest temperatures recorded by the sensor at 6 cm are 51.5 °C and 41.2 °C respectively, whereas such temperatures respectively reach 43.3 °C and 34.39 °C at 20 cm. Such comparison confirms that the maximum temperature is recorded at 0 degree, whereas the minimum temperature is recorded at 135 degrees. In other words, if a WSN is implemented for fire detection under windy conditions, the time for fire detection will be the fastest for wind direction of 0 degree and the slowest for wind direction of 135 degrees. For wind directions of 45 or 90 degrees, the collected temperatures depend on the distance between the sensor node and the heat. At 6 cm, the temperature registered for wind direction of 45 degrees is higher than that registered for wind direction of 90 degrees, which is not the case at 20 cm. As a result, for wind directions of 45 or 90 degrees, the time for fire detection depends on the distance between the heat and the WSN. An important lesson learned is that the wind direction has an impact on the temperatures collected by the sensors and can be an important parameter for detecting fire on time using a WSN. Note that at 180 degrees, the registered temperatures are not the lowest for both experiments.

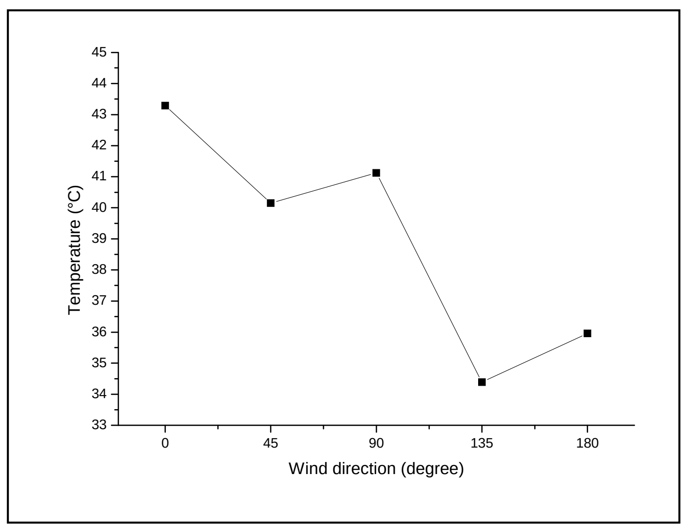

Figure 5. Temperature variation in function of the wind direction at 20 cm.

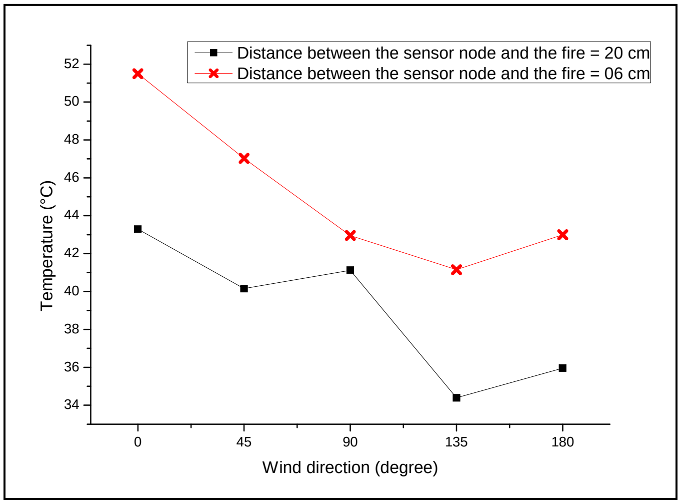

Figure 6. Comparison of temperatures recorded for two different distances.

IV. CONCLUSION AND FUTURE WORK

In this paper, a set of experiments were presented for assessing the impact of wind direction on the temperatures recorded by a WSN. In this context, a real heat was generated using an oven, whereas the presence of wind was simulated using a fan that had different orientations. Several applications were implemented in the sensor nodes for recording environmental temperatures and sending such information to a sensor gateway. Results analysis shows that the wind direction has an impact on the recorded temperatures. More specifically, if a WSN is implemented for fire detection under windy conditions, the time for fire detection can be the fastest for wind direction of 0 degree and the slowest for wind direction of 135 degrees. Future work should focus on methods and algorithms that consider the impact of both wind direction and wind speed on fire detection using a WSN.

REFERENCES

[1] D. Puccinelli and M. Haenggi. “Wireless Sensor Networks: Applications and Challenges of Ubiquitous Sensing,” IEEE Circuits and Systems Magazine, vol. 5, pp. 19-31, 2005.

[2] D. M. Doolin and N. Sitar, “Wireless Sensor for Wildfire Monitoring,” Smart Structures and Materials 2005: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems, SPIE, Volume 5765, pp. 477-484, San Diego, USA, 2005.

[3] J. Glasa and L. Halada, “Envelope Theory and its Application for a Forest Fire Front Evolution,” Journal of Applied Mathematics, Statistics and Informatics, No. 1, pp. 27-37, 2007.

[4] M. I. Asensio and L. Ferragut, “On a Wildland Fire Model with Radiation,” Int. Journal for Numerical Methods in Engineering, vol. 54, pp. 137-157, Feb. 2002.

[5] Z. Nauman, S. Iqbal, M. I. Khan, and M. Tahir, “WSN-Based Fire Detection and Escape System with Multi-modal Feedback,” MCSS 2011, CCIS, vol. 149, pp. 251–260, 2011.

[6] P. Roy, S. Bhattacharjee, S. Ghosh, S. Misra, and M. S. Obaidat, “Fire Monitoring in Coal Mines Using Wireless Sensor Networks,” Performance Evaluation of Computer & Telecommunication Systems (SPECTS), the Hague, Netherlands, Jun. 2011.

[7] E. S. Manolakos, E. Logaras, and F. Paschos, “Wireless Sensor Network Application for Fire Hazard Detection and Monitoring,” SENSAPPEAL, vol. 29, pp. 1–15, Athens, Greece, Sep. 2009.

[8] L. Shixing, X. Wujun, and Z. Yongming, “Research and Implementation of WSN in Fire Safety Applications,” IEEE 6th International Conference on Wireless Communications, Networking and Mobile Computing, Chengdu, China, Sep. 2010.

[9] R. Beckwith and P. Bowen, “Report from the Field : Results from an Agricultural Wireless Sensor Network”, IEEE 29th Conference on Local Computer Networks, pp. 471-478, Tampa, Florida, Nov. 2004.

[10] G. D. Richards, “An Elliptical Growth Model of Forest Fire Fronts and its Numerical Solution”, International Journal for Numerical Methods in Engineering, vol. 30, pp. 1163-1179, 1990.

[11] J. Lloret, M. Garcia, and S. Sendra, “A Wireless Sensor Network Deployment for Rural and Forest Fire Detection and Verification”, Special Issue State-of-the-Art Sensors Technology in Spain, vol. 9, pp. 8722-8747, Valencia, Spain, 2009.

[12] S. Osman, J. E. Penha, and F. Nakamura, “Fusing Light and Temperature Data for Fire Detection,” IEEE Symposium on Computers and Communications, Riccione, Italy, pp. 107-112, Jun. 2010.

[13] L. Yu, N. Wang, and X. Meng, “Real-time Forest Fire Detection with Wireless Sensor Networks,” International Conference on Wireless Communications, Networking and Mobile Computing (WIMOB), vol. 2, pp. 1214–1217, 2005.

[14] S. Smolau, “Evaluation of the Received Signal Strength Indicator for Node localization in Wireless Sensor Networks”, M.Sc. thesis, Laval University, Quebec, 2008.

[15] Moteiv. “Tmote Sky Ultra Low Power IEEE 802.15.4 Compliant Wireless Sensor Module Datasheet,” 2006.

[16] TinyOS: An open-source OS for the wirless embedded sensor networks. http://www.tinyos.net. [Retrieve: July, 2012].