Forest Fire Detection with Wireless Sensor Networks

Çağdaş Döner*, Gökhan Şimşek, Kasım Sinan Yıldırım and Aylin Kantarcı
Computer Engineering Department, Ege University

Abstract- : In this paper, we present a wireless sensor network for detection of forest fires. We first describe the architecture of the forest fire detection system. We then present our implementation on Genetlab Sensenode platform using TinyOS 2.1.

Index Terms- Wireless Sensor Networks, Forest Fire Detection, Distributed Algorithms, Localization, Spanning Tree

I. Introduction

Forest fires generally occur in wild areas due to human carelessness and change in air conditions. They cause threats to the ecosystem and may result in human and animal deaths. Consequently, forest fires must be detected early to prevent greater damages.

Wireless sensor networks (WSN) consist of tiny, cheap and low-power sensor devices that have ability to sense the environment. Spatially separated sensor nodes collaboratively collect, process and disseminate environmental data through wireless broadcast medium by exchanging messages. Immediate notification of the fire is the most critical issue in a forest fire detection systems. WSN can provide real-time fire detection with high accuracy [1][2].

In this paper, we design and evaluate a wireless sensor network for early detection of forest fires. The general architecture of our system is presented in Section II. We present the implementation details in Section III. Finally, we offer our conclusions in Section IV.

II. A Wireless Sensor Network for Forest Fire Detection

Our forest fire detection system consists of nodes deployed random in a forest area, as seen in Fig. 1. Each node is equipped with a temperature sensor. Nodes periodically sense the environment to decide if there is an emergency situation or not. When a significant change in temperature is detected by some sensor nodes, they broadcast packets which contain their measurements. These packets are received by the nodes which are in the broadcast region of the sender and relayed to the node that acts as the *base station*. Base station is a special node which is connected to a computer and forwards the received messages to the serial port. An application listens to the serial port and keeps track of the network.

Since the nodes are deployed randomly, we do not have information about their coordinates. In addition, a topological infrastructure is needed for the flow of messages from nodes to the base station. We used a simple *localization* to get coordinates of the nodes and a *distributed spanning tree* [3] for the messaging infrastructure. We will now describe these procedures in detail.

Fig. 1 Forest Fire Detecton System

A. A Distributed Spanning Tree Construction Protocol

We used a distributed spanning tree protocol to collect and route data from nodes to the base station. During the execution of the protocol, nodes cooperate together by exchanging messages. By considering messages received from their neighbors, each node chooses a parent node to which they will forward observations from the environment.

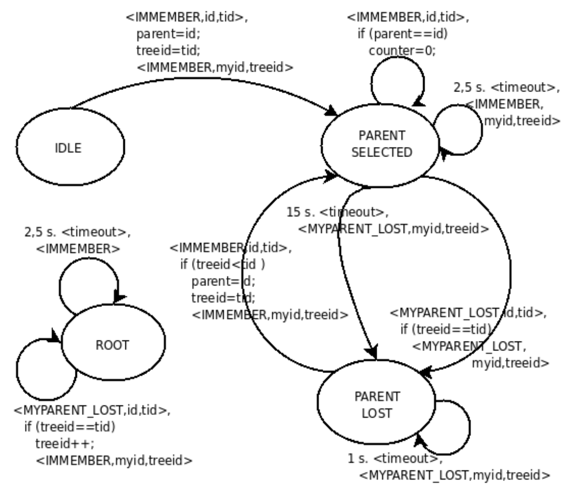

Fig. 2 FSM For The Spanning Tree Protocol

The finite state machine (FSM) for the protocol is shown in Fig. 2. At the beginning of the protocol, the root node broadcasts "*I am member*" message with a *tree id*. When a node in "*idle*" state receives this message, it changes its state to "*parent selected*" and broadcasts "*i am member*" with its own node id. This chain continues until all nodes in the system select their parents. In addition, all nodes periodically broadcast "*i am alive*" messages to state that they are not crashed or run out of battery.

When a node in "*parent selected*" state node can't receive any message from its parent for 15 seconds, it sets its state to "parent lost" and broadcasts a "my parent lost" message. Any node which receives this message changes its state to "*my parent lost*" and forwards this message until it reaches to the root node. When the root receives a "*my parent lost*" message, it increments the tree id and broadcasts "*i am* *member*" message with this tree number which restarts spanning tree construction again. Nodes receiving this message compare their current tree id with the received tree id and set their parent if the received value is greater.

B. *Simple Localization*

We used a self-localization technique which allows nodes to determine their coordinates using the information periodically broadcasted by the *anchor* nodes. The anchor nodes are deployed manually and know their own coordinates. Each sensor node needs the coordinate information of three anchor nodes. After receiving three anchor messages, each node performs a trilateration [4] to calculate its own coordinate. Fig. 3 shows the sensor network with anchor nodes.

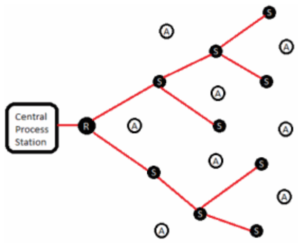

Fig. 3 Anchor nodes

Nodes can estimate the distance between anchor nodes and itself using Received Signal Strength Indicator (RSSI) values. With three received coordinate and RSSI values, the nodes in the system solve the following linear matrix equation [5]:

$$2\begin{bmatrix} x_3 - x_1 & y_3 - y_1 \\ x_3 - x_2 & y_3 - y_2 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} (r_1^2 - r_3^2) - (x_1^2 - x_3^2) - (y_1^2 - y_3^2) \\ (r_2^2 - r_3^2) - (x_2^2 - x_3^2) - (y_2^2 - y_3^2) \end{bmatrix}$$

$(x_1, y_1), (x_2, y_2), (x_3, y_3)$: received coordinate values

r_1, r_2, r_3: received RSSI values

x, y: coordinates to be calculated

III. IMPLEMENTATION

The hardware platform used for the implementation of the forest fire detection system is Genetlab Sensenode v.1.3 [6] sensor nodes shown in Figure 4. The Sensenode platform includes 16- bit low-power MSP430 micro-controller having 10kB RAM, 48kB program flash and 1024kB external flash memory. The Chipcon CC2420 radio chip provides a 250kbps data rate at 2.4 GHz frequency. SHT11 temperature sensor is used to gather temperature values from the environment.

Fig. 4 Genetlab Sensenode v.1.3

The system is implemented using TinyOS [7] operating system and nesC programming language. *TinyOS* is an open-source operating system designed for wireless embedded sensor networks. It features a component-based architecture. The component library of TinyOS includes network protocols, distributed services, sensor drivers, and data acquisition tools. However, we implemented our own protocols and did not used protocol libraries provided by TinyOS.

A. *TinyOS Implementation*

We have mainly used components of TinyOS to broadcast data, to get RSSI values and to gather temperature values from the environment. For broadcasting and receiving data, we used *ActiveMessageC*, *AMSenderC* and *AMReceiverC* components. *TempSensorC* component provides *Read* interface to get reliable data from SHT11 temperature sensor.

At each sensor, we maintained a 4x3 array for storing RSSI values and coordinates received from anchor nodes. *CC2420Packet* component provides us an interface for obtaining the strength of the received signal.

B. *Java Application*

We implemented a Java application for monitoring the system and for the visualization of the gathered data. Our application listens to the serial port and gets the coordinates and temperature values of the nodes which are forwarded to the serial port by the base station. The application marks the coordinates of the disaster on a pre-loaded map, when a critical temperature value is received from a node.

IV. CONCLUSION AND FUTURE WORK

In this paper, we presented the design and implementation of a wireless sensor network system for detecting forest fires. Nodes in the system periodically sense the environment and send their measurements to the base station through the spanning tree. The coordinates of the nodes are calculated using a simple localization procedure. A java application interprets the received data to detect coordinates disaster. In order to get precise timing information of the fire event and to logically order the measurement times, we plan to improve our system by adding a time synchronization protocol in the future.

V. REFERENCES

[1] Hedeeda, M.: Forest Fire Modeling and Early Detection using Wireless Sensor Networks. Technical report CMPT 2007, Faculty of Applied Sciences, Simon Fraser University, Canada (2007).

[2] Majid Bahrepour, Nirvana Meratnia, Paul Havinga: Automatic Fire Detection: A Survey From Wireless Sensor Network Perspective. Pervasive Systems Group, University Of Twente, 2008.

[3] http://en.wikipedia.org/wiki/Spanning_tree

[4] Yingshu Li,My T. Thai,Weili Wu , *Wireless Sensor Networks and Applications*, Springer, 2008,ISBN 978-0-387-49591-0

[5] Holger Karl and Andreas Willig: *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & Sons Ltd., 2005, ISBN: 0-470-09510-5

[6] http://www.genetlab.com/

[7] http://www.tinyos.net