Wireless Local Positioning System for Unmanned Surface Vehicles

Abdelrhman A. Eltohamy and Ali S. Ali
MTC, Egypt, thamy.abdo@yahoo.com, Lol.7egazy@gmail.com

Supervisor: Ahmed E. Abdalla, Prof.
MTC, Egypt, elsaid.mtc@gmail.com

Abstract*– The instructions give the basic guidelines for Wireless Local Area Networks (WLAN) can be constituted for localizing wireless-enabled clients where GPS and GLONASS are unattainable due to multipath and signal blocking. Advances in unmanned surface vehicles (USVs) missions require a reliable navigation system to be controlled autonomously. A wireless sensors network becomes a necessary local positioning service (LBS) in a cooperative USVs. Many techniques are used to estimate client position in a wireless network. They are based on the characteristics of the received wireless signals: power, time or angle of arrival. In this paper a survey of positioning techniques and location based services in wireless networks is discussed. A practical application method of localization with use of WiFi of access points is applied in real time. A number of algorithms and techniques based upon different characteristics and properties of sensor nodes have already been proposed for this purpose. Results of system operation are considered with different arrangements of access points in a given space.

Keywords*— Wireless Local Area Networks, GPS, WiFi networks, unmanned surface vehicles, local positioning service.

I. INTRODUCTION

Hasty advance in the field of wireless data communication has establish a significant applications to embedded radio frequency (RF) transceivers to afford location based services (LBS) to users, in addition to their main function of communication. Due to drawbacks in using Global Navigation Satellite Systems (GNSS) or having to manually setting sensor positions, extracting the position information by means of the network itself has been extensively studied in the literature. In this approach, it is commonly assumed that there are a few sensor nodes with known positions, called reference nodes, and some type of measurements are taken between different nodes.

Certain applications of wireless sensor networks require that the sensor nodes should be aware of their position relative to the sensor network. For it to be significant and to be of value, the data such as temperature, humidity and pressure gathered by sensor nodes must be ascribed to the relative position from where it was collected. For this to happen, the sensor nodes must be aware of their positions. The literature has come to term this problem of location or position estimation of sensor nodes simply as localization. The term localization has earlier been used in robotics where it is used to refer to determination of location of a mobile robot in some coordinate system. Under certain circumstances, the nodes should not only by aware of their position but also the direction or orientation relative to the network.

Currently WiFi networks are so pervasive that can without difficulty build a network consisting of a large number of devices. At present almost any corporate client has WiFi coverage, therefore the infrastructure of this wireless network can be used for performing a number of tasks, in particular, for local positioning in the organization. Accuracy of such systems will depend on density of points of access tied to specific points in the plan of buildings, constructions and territories [1].

Positioning of client objects in WiFi networks can be mainly used for automation of location placement and data collecting for methodical analysis. A client-server Wi-Fi communication (ad hoc network) is built between ESP8266 NodeMCU boards for positioning and automation. For simulating the control of USV an H-Bridge circuit is configured based on digitally controlled four solid state relay circuit to control the direction and the speed of DC motors. Microcontroller sending it the appropriate control signals with the PWM.

The paper is organized where, in section-II review the classification of localization algorithm. Section-III, gives the details of distance estimation. While in section-IV, explore the details of positioning in wireless networks.
Section-V simulation and experimental work. Work conclusion in this work is given in Section VI.

II. Classification of Localization Algorithms

Majority of the existing localization algorithms may be classified as ranged-based or range-free contingent upon whether the algorithm uses distance estimation or fusion of other data for estimating the node locations. Range-based algorithms usually use sensor field geometry information to determine node locations. Communication between sever (beacon) nodes and client (dumb) nodes are used to determine their relative placement or the angles of a triangle formed by the beacons nodes lead to determine node location. The term lateration is used when distance is used to determine node location. The term angulation is used when angle information, to determine node location. For node localization in a plane, precise distance measurements from at least three beacon nodes are required and we use trilateration for position estimation of a node. Intersection of three circles around the three beacon nodes gives a single point as position of the node as is shown in Fig. 1.

The same technique can be extended to three-dimensional space by the addition of a fourth beacon node. However, in actual practice, distance measurements are seldom precise and intersection of three circles may result in more than one point. The scheme may be improved by employing more than three beacon nodes for a plane and we then use multilateration to calculate the node position.

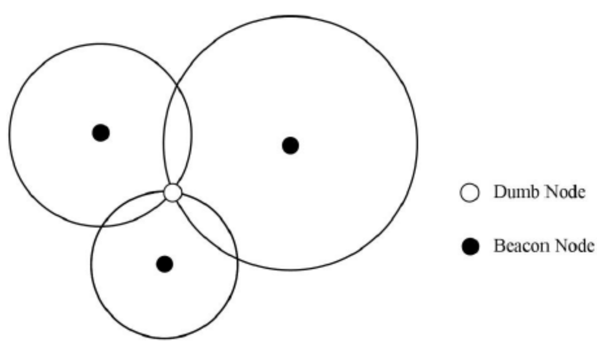

Fig. 1. Trilateration – Intersection of three circles around three beacon nodes gives position of the client node.

Moreover there is what is called range-free localization algorithms where a node determines its position merely by finding the beacon nodes in its proximity and for this reason, range-free algorithms are also termed as proximity-based or connectivity-based algorithms. Such algorithms usually provide coarse-grained localization. However, with sufficient number of beacon nodes with overlapping transmission regions, a more accurate localization is possible. Range-free algorithms are robust against the fluctuations of the wireless channel as the decision whether a node is in proximity of another node is based upon connectivity information sampled over a long period of time. Hence, short and temporary variations in the wireless channel do not affect the accuracy of location estimation. A range-free algorithm calculates the position without having to find the distance between the sensor nodes. A range-based localization algorithm is more accurate but has major computational cost and usually additional hardware and hence increased energy requirements. On the other hand, a range-free algorithm is less accurate but does not require additional hardware and has smaller computational overhead.

III. DISTANCE ESTIMATION

Distance estimation between two nodes is an important function performed by range-based algorithms. A range-based algorithm estimates the position of a sensor node by using the distance information between the nodes which, in turn, is calculated using some physical measured quantity.

The distances between client nodes and the beacon nodes are usually determined by adding some additional hardware to the nodes or by using the existing radio communication facility on the sensor nodes. Certain characteristics of wireless communication between client and beacon nodes are determined by the distance between them. If these characteristics are quantified and measured at the receiving sensor node, these can be used to estimate the distance between the nodes. The characteristics generally used for this purpose are [6-9]:

- Received Signal Strength Indicator (RSSI)
- Time of Arrival (ToA)
- Time Difference of Arrival (TDoA)
- Angle of Arrival (AoA)

A. Received Signal Strength Indicator (RSSI)

The strength of the received signal can be used to estimate the distance between the transmitter and receiver. The distance can be calculated using the following information

- Transmitted power of the signal
- Received power of the signal
- Path loss model

Using these three parameters, power of the received signal P^{ij}_R transmitted by node *i* and received at node *j* at time *t* can be expressed as:

$$P^{ij}_R(t) = P^i_T - 10\eta log(d_{ij}) + X_{ij}(t) \qquad (1)$$

In this equation:

$P^{ij}_R(t)$ is power of the received signal at receiver node *j* transmitted by node *i* at time *t*.

P^i_T is transmitted power of signal transmitted by node *i*.

η is attenuation constant, value of which depends upon the surroundings of the receiver node *j*.

d_{ij} is distance between transmitter node *i* and receiver node *j*.

$X_{ij}(t)$ is uncertainty factor or channel model whose value depends upon multipath fading and shadowing.

Equation above can be solved for distance d_{ij} between beacon node *i* and receiver node *j* as all other parameters are known.

The fundamental assumption in a localization scheme based upon RSSI is that the signal suffers from the same amount of attenuation for the same distance travelled. However, in actual practice, this is not always the case due to factors such as multipath fading, fast fading and shadowing. Savarese, Rabaey & Langendoen [2] have reported that ranging errors of the order of ±50% are possible even when both the transmitter and receiver nodes are stationary. This problem can be remedied, to some extent, by taking more measurements. In addition, statistical techniques may be employed to filter out incorrect values as suggested by Ward, Jones & Hopper [3].

If there are obstacles between the beacon node and client node such that line of sight communication between them is not possible, the later receives signal transmitted by the beacon node after reflection from the surroundings. As a result, the signal suffers far greater attenuation than the case of line of sight communication between the two nodes, and as a consequence, the RSSI value is not a true indicator of the distance vector between the two nodes and yields a distance estimate which is much greater than the actual distance [4,5]. The resultant error cannot be corrected by increasing the number of measurements since the additional measurements are still based upon RSSI values of signals which are received after reflection from the surroundings.

B.Time of Arrival

This technique of distance estimation uses the following relationship that relates the distance travelled by a signal to the time taken provided that the speed of propagation is known.

$$d = v \times t \qquad (2)$$

where *d* is distance, *v* is speed of the signal and *t* is time taken by the signal to travel the distance *d*. Therefore, if the time taken by a signal to propagate from the beacon node to the client node, which is called *time of arrival* or *time of flight* is measured and speed of propagation of the signal is known, the distance and hence position of the client node can be calculated.

There are two variations of the time of arrival technique:

- One-way time of arrival
- Two-way time of arrival

Furthermore, either of the above techniques may use an RF signal or an ultrasonic pulse for distance estimation.

A. One-way time of arrival

In one-way time of arrival technique, the propagation time of one-way trip of the signal from the beacon node i to client node j is measured. This is given by the difference between the sending time ti at beacon transmitting node and receiving time tj at the receiving client node. The distance d_{ij} between the two nodes i and j is then given by:

$$d_{ij} = v \times (t_j - t_i) \qquad (3)$$

With this approach, the receiver node calculates its position in a secure manner without disclosing its location information to the transmitting node, and hence is also termed as *passive time of arrival* localization. Obviously, transmitting node is usually a beacon node and the receiving node is the client node. As stated earlier, an RF signal or ultrasonic pulse can be used for distance estimation using this technique.

It is to be noted that for the one-way time of arrival technique to work the receiver must know the time of transmission of the signal. In the case of an RF signal, the transmitting node can embed this information in the beacon signal that it sends to the receiver. However, RF signals travel at a very high speed, which is almost equal to the speed of light i.e. 3×10^8 m/s. Their use for distance estimation using time of flight requires extremely accurate and stable clocks. The requirement of highly stable and accurate clocks and precise timing measurement means addition of complex and costly hardware to the sensor node and necessitates usage of time synchronization algorithm along with localization algorithm so as to increase size and processing need and thereby energy consumption.

B. Two-way time of arrival

In the two-way approach, the receiver node sends the signal back to the transmitter node which measures the round-trip time for distance estimation between transmitter and receiver. Suppose sensor node i transmits signal at its local time ti1. It reaches sensor node j at its local time tj1. After some delay, the sensor node j sends the signal back at its local time tj2. The signal is received back at node i at its local time ti2. This is illustrated in Fig. 2.

Total round-trip time including delay = $t_{i2} - t_{i1}$

Delay suffered at $j = t_{j2} - t_{j1}$

Actual roundtrip time = $(t_{i2} - t_{i1}) - (t_{j2} - t_{j1})$

One-way time of flight = $((t_{i2} - t_{i1}) - (t_{j2} - t_{j1}))/2$

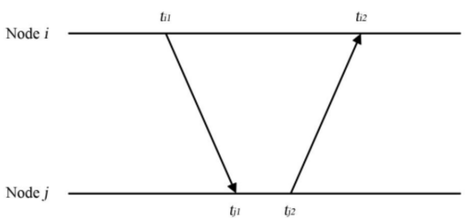

Fig. 2 Signal propagation times in two-way time of arrival distance estimation technique.

Hence, the distance *d* between the two nodes is given by:

$$d = ((t_{i2} - t_{i1}) - (t_{j2} - t_{j1}))/2 \times v \qquad (4)$$

It should be noted that node *i* subtracts its local time $t_{i2} - t_{i1}$ to get the total roundtrip time. Similarly, node *j* subtracts its local time $t_{j2} - t_{j1}$ to get the processing delay. As both nodes have to process only their respective local times, no time synchronization between the nodes is necessary. This saves extra hardware cost and energy required for time synchronization.

With the two-way approach, the beacon node, which is the transmitting node, has to carry out processing for distance and position estimation. The result is then sent back to the client node. As there are a large number of client nodes per beacon node, a particular beacon node may have to process localization for many client nodes. This will increase processing overhead on the beacon nodes by manifold. Furthermore, communication overhead is also involved as the beacon node has to send the result back to the client node.

With the one-way time of arrival approach, the beacon nodes do not have any processing overhead as localization processing is carried out by the client nodes. In this way, the processing task is uniformly distributed amongst all the client nodes, and beacon nodes are also free to send timely beacons to the client nodes.

C. Time Difference of Arrival (TDoA)

Time difference between the receiving of two signals at a node is easier to measure compared to time of arrival of a signal. This time difference information can then be used to estimate the distance between the two nodes. Advantage of using time difference instead of time of arrival is that errors in time difference measurement are tolerable and do not have a pronounced effect on the accuracy of estimation of distance between two nodes. As a result, the hardware required for time measurements is less complex and less costly and hence the method is also efficient in terms of energy consumption.

The TDoA techniques can be classified into two main categories: Multi-node TDoA ,Multi-signal TDoA

D. Multi-node TDoA

At least three beacon nodes B1, B2 and B3 transmit signals at exactly the same time. Time differences amongst the arrival of these three signals at the receiving client node D are measured. The difference from a pair of beacon nodes, say B1 and B2, defines a branch of hyperbola on which the client node D is located. Similarly, difference from the pair of beacon nodes B2 and B3 will again give branch of a second hyperbola. The receiving client node should lie on this second hyperbola as well. The point of intersection of the two hyperbolas gives the location of the client node D.

It should be noted that the nodes should be time synchronized with stable clocks for them to be able to transmit the beacon signals at exactly the same time. This type of TDoA is quite old and was used in the classical long range navigation systems such as LORAN.

E. Multi-signal TDoA

In the time of arrival (ToA) technique using RF signals, sophisticated hardware is needed for precise measurements of time and the nodes should also be time synchronized. One way to alleviate this problem is the use of an ultrasonic signal along with an RF signal. The beacon transmitting node *i* transmits the two signals simultaneously or after some fixed time interval *ti2 – ti1* as shown in Fig. 3.

Due to a large difference in their propagation speeds, the RF signal is received first by the receiving client node *j*, which records the arrival time t_{j1} of the RF signal. As the speed of ultrasonic signal is very small compared to the speed of RF signal, this time of arrival of RF signal is treated as the time of transmission of the ultrasonic signal. After receiving RF signal, the receiving client node prepares itself to receive the ultrasonic signal. Time of arrival t_{j2} of ultrasonic signal is also recorded. If speed of the ultrasonic signal is u, an estimation of the distance d between the beacon transmitting node and the receiving client node is given by:

$$d \approx (t_{j2} - t_{j1}) \times u \quad (5)$$

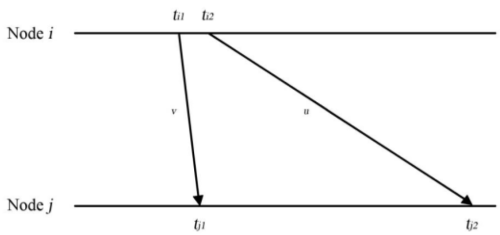

Fig. 3. Signal propagation times in multi-signal TDoA distance estimation technique.

It should be noted that sender and receiver clocks need not be synchronized because the RF signal provides for an indirect synchronization mechanism between the transmitting and receiving nodes. Disadvantage of this scheme is that the nodes must be equipped with separate pieces of hardware for transmission, reception and processing two different kinds of signals. For the ultrasonic signals, the nodes must be equipped with microphones and speakers. This results in increased cost, size and energy consumption in the sensor nodes. Instead of RF and ultrasonic signals, any two signals with large difference in their speeds can be used for multi-signal.

Position estimation using multi-signal TDoA is quite accurate compared to other ranging techniques in the order of a few centimeters. However, this is only possible under line of sight conditions. Under non line of sight conditions, both RF and ultrasonic signals suffer from propagation delays, and position estimation using multi-signal TDoA is error prone. In particular, ultrasonic waves may suffer from attenuation due to scattering and diffraction resulting from the obstacles and atmospheric effects such as temperature, pressure, humidity and turbulence.

F. Angle of Arrival

The direction of arrival of a signal at the client node can also be used to estimate its position. The direction of a received signal can be determined by measuring the angle it makes with some reference direction or orientation. Alternatively, the angle between the client node and the beacon node may be measured. For the localization of a client node using this technique, angles of arrival from a minimum of three beacon nodes are measured. Position information of three or more beacon nodes along with the three angles of arrival can be used to estimate the location of the client node.

The angle of arrival can be measured using directional antennas, a special configuration of antenna arrays or a combination of both. When using directional antennas, these can be mounted on the beacon nodes. To serve multiple client nodes, a directional antenna mounted on a beacon node rotates about its axis thereby transmitting beacon signals in all directions. A client node may use a similar directional antenna configuration to receive the beacon signals. Alternatively, client nodes can also use special configuration of antenna arrays to receive and measure the angle of arrival of a beacon signal. When an antenna array is used, antennas in the array are placed at known separation. The difference of time of arrival of the wave front at different antennas is used to estimate the direction from which the signal arrived.

Practical use of this technique is limited due to the complexities of deployment of special antennas. For example, mounting rotating directional antennas on tiny nodes is problematic and the rotating components are more prone to failure. Similarly, if an antenna array configuration is used, antennas in the array must be placed specific distance apart which is again a difficult proposition considering the tiny sizes of sensor nodes. Moreover, a greater accuracy of angle measurement is achieved only when separation distance between antennas in the array is small. However, with smaller separation distance, more sophisticated and precise hardware is needed for time difference measurements. Furthermore, shadowing, multipath fading and non line of sight conditions introduce a large amount of error in the estimated position which is more than same kind of errors in other similar techniques e.g. RSSI, ToA and TDoA. Due to these reasons, angle of arrival is considered less of a choice for localization in sensor networks.

After distances have been estimated by using one of the techniques discussed above, multilateration is employed to estimate the position of a client node. Obviously these techniques form the basis of range-based localization algorithms. Range-free algorithms do not use measurements to estimate distances and for localization. Instead, range-free algorithms analyze the connectivity information of neighbor nodes to deduce position information.

Apart from range-based and range-free methods, still another possible technique of localization is signal pattern matching. A database of unique signal signatures for all possible locations is created by using some property of radio signals. A client node localizes itself by comparing the pattern of received signals with the stored signal signatures. For example, a fixed number of beacon nodes may be deployed in the sensor field and RSSI values at each possible location may be calculated and stored in a database so as to serve as location signatures. This database can then later be used for localization after actual sensor nodes are deployed in the sensor field. It should be noted that other signal characteristics instead of RSSI values, such as multipath pattern of a signal arriving at a given location can also be used to create unique signatures for each location. Similarly, multiple signal characteristics may be combined to develop the signature database. This information can then be used to locate the position of a node. However, due to its very nature, this technique is not suitable for networks which are ad hoc and may have a dynamically changing topology, which is the case with sensor networks.

IV. POSITIONING IN WIRELESS WIFI NETWORKS

Positioning in wireless networks can be carried out in several ways:

1. The method of recognition of a template demanding a preliminary research of the network and control of the system.
2. The method of using the coordinates of the nearest access point which allows detecting the presence of the client.
3. The triangulation method involving determination of force of a WiFi signal and calculation of a possible arrangement concerning each visible point of access.

An angulation or positioning with definition of the angle of the incoming signal.

The method of a triangulation is rather simple and effective, and it doesn't demand high costs to apply it as well, therefore it is used in this system. The system consists of several points of access, a WiFi-tag and a device for information collection and data management as shown in Fig. 4. The system operates as follows: the device (tag) searches for the given points of access and measures the power of the accepted signal, then sends data to the device where the location is calculated with some accuracy in the set system of coordinates [9,10].

Data on the control unit are transmitted "by air" through WiFi technology and can be easily intercepted. For their closing the known encryption methods in WiFi of network can be used (if it is necessary).

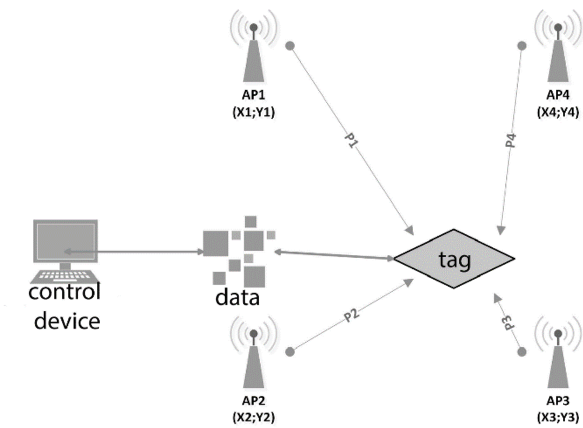

Fig. 4. The principle of operation of the local positioning in WiFi network.

The coordinates were calculated by means of the algorithm Weighted centroid (the center of masses) which computes the agent's coordinate as a linear combination of coordinates of points of access taking into account the power of signals as the characteristic of weight [9,10]. The way this method calculates the AP position determines the situations in which the method provides good location estimations: If there are measurements taken around the real AP position or the measurements are close to the AP [10,11].

To calculate the location the following formulae are used:

$$\begin{cases} X_o = \sum_{i=1}^{N} \mu_i X_i \\ Y_o = \sum_{i=1}^{N} \mu_i X Y_i \end{cases} \quad (6)$$

$$\mu_i = \left(P_i \sum_{j=1}^{N} \frac{1}{P_j^2} \right) \quad (7)$$

where X_o and Y_o– required coordinates, Xi and Yi – coordinates of i-th point of access, Pi – the power of a signal of i-th point of access, Pj – the power of a signal of j-th point of access, μ_i-characteristic of weight, N – quantity of points.

The characteristic of weight of a signal μ_i from i-that access point is calculated by equation (7). After each scanning it is necessary to calculate the characteristic of weight on equation (7). After that tag coordinate on equation (6) is calculated. As a tag using the SP8266 Node MCU WiFi (see Fig. 5).

Fig. 5. ESP8266 NodeMCU.

In that this microcontroller has GPIO interface, connection to them of different sensors make the device is possible. Such sensors can serve for determination of falling, violation of integrity of an object, and also to collect object other information. Data transfer is carried out on the WiFi network.

The board is programmed so that at first to scan WiFi of a network, and then to transfer them to the server. After successful transmission the device starts over again scanning. The broadcast a WiFi-signal at a frequency of 2.4 GHz.

V. Simulation and Experimental work

In this part of work we develop the different configurations of ESP8266 as client-server and switching control of the H-Bridge circuit. The algorithm based on *TDoA as* distance measure is developed.

A. ESP8266 Client-Server Wi-Fi Communication Between Two Boards (NodeMCU)

A Wi-Fi communication (HTTP) between two ESP8266 NodeMCU boards is configured to exchange data without the need to connect to the internet (no need to a router). The ESP8266WebServer library allows you run an ESP8266 as a basic webserver and access point. ESP8266 is set as an Access Point (Server) and another ESP8266 as a Station (Client), see Fig. 6. Then, the server and the client will exchange data (sensor readings) via HTTP requests, then program the ESP8266 boards using Arduino IDE.

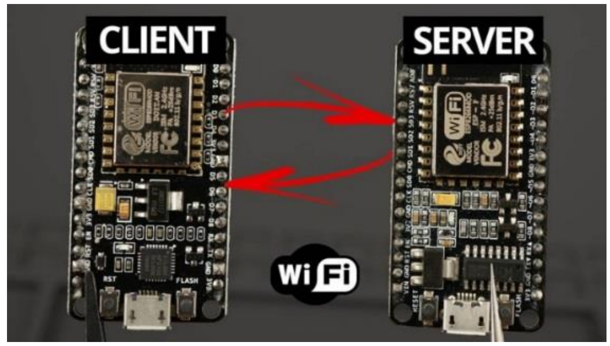

Fig. 6 ESP8266 Client-Server Wi-Fi Communication Between Two Boards

The Arduino program to configure ESP8266 as server and client are given in Fig. 7 and Fig. 8 respectively.

```
// Sever Node
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <U8g2lib.h>


const char *ssid = "poopssid";
const char *password = "pingu4prez";

ESP8266WebServer server(80);

void handleSentVar() {
 Serial.println("handleSentVar function called...");
 if (server.hasArg("sensor_reading")) { // this is the variable sent from the client
  Serial.println("Sensor reading received...");

  int readingInt = server.arg("sensor_reading").toInt();
  char readingToPrint[5];
  itoa(readingInt, readingToPrint, 10); //integer to string conversion for OLED library

  Serial.print("Reading: ");
  Serial.println(readingToPrint);
  Serial.println();
  server.send(200, "text/html", "Data received");
 }
}

void setup() {
 delay(1000);
 Serial.begin(115200);
 Serial.println();
 Serial.print("Configuring access point...");

 WiFi.softAP(ssid, password);

 IPAddress myIP = WiFi.softAPIP();
 Serial.print("AP IP address: ");
 Serial.println(myIP);
 server.on("/data/", HTTP_GET, handleSentVar); // when the server receives a request with /data/ in the string then run the handleSentVar function
 server.begin();
 Serial.println("HTTP server started");

}

void loop() {
 server.handleClient();
}
```

Fig. 7 Sever configuration program

```
//Client
#include <ESP8266WiFi.h>
#include <U8g2lib.h>

const char *ssid = "poopssid";
const char *password = "pingu4prez";

const int analogInPin = 0;  // Analog input pin that the potentiometer is attached to
int sensorValue = 0;        // value read from the potentiometer
int outputValue = 0;        // value sent to server

void setup() {
  Serial.begin(115200);
  delay(10);

  // We start by connecting to a WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  /* Explicitly set the ESP8266 to be a WiFi-client, otherwise, it by default,
     would try to act as both a client and an access-point and could cause
     network-issues with your other WiFi-devices on your WiFi-network. */
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  Serial.begin(115200);
//  u8g2.begin();
//  u8g2.setFont(u8g2_font_logisoso62_tn);
//  u8g2.setFontMode(0);    // enable transparent mode, which is faster
}

void loop() {

  outputValue = map(sensorValue, 3, 1023, 0, 999);

  // print the results to the Serial Monitor:
  Serial.print("sensor = ");
  Serial.print(sensorValue);
  Serial.print("\t output = ");
  Serial.println(outputValue);

  char intToPrint[5];
  itoa(outputValue, intToPrint, 10); //integer to string conversion for OLED library

  // Use WiFiClient class to create TCP connections
  WiFiClient client;
  const char * host = "192.168.4.1";
  const int httpPort = 80;

  if (!client.connect(host, httpPort)) {
    Serial.println("connection failed");
    return;
  }

  // We now create a URI for the request
  String url = "/data/";
  url += "?sensor_reading=";
  url += intToPrint;

  Serial.print("Requesting URL: ");
  Serial.println(url);

  // This will send the request to the server
  client.print(String("GET ") + url + " HTTP/1.1\r\n" +
               "Host: " + host + "\r\n" +
               "Connection: close\r\n\r\n");
  unsigned long timeout = millis();
  while (client.available() == 0) {
    if (millis() - timeout > 5000) {
      Serial.println(">>> Client Timeout !");
      client.stop();
      return;
    }
  }

  Serial.println();
  Serial.println("Closing connection");
  Serial.println();
  Serial.println();
  Serial.println();

  delay(500);
}
```

Fig. 8 Client configuration program

B. Controlling a DC Motor using an H-Bridge

An H-Bridge is made up of four solid state relay switches: two in series, and two in parallel, with the load placed in between the switches. In this configuration the circuit takes an "H" shape see Fig. 9. An example of the digital switching waveform as PWM to control the motor speed is shown in Fig. 10. The wiring connection of ESP8266 with four channel SSR is given in Fig. 11, where the control pins are D0 , D1, D2 and D3.

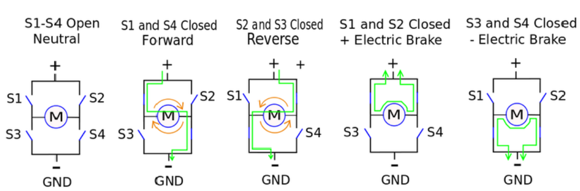

Fig. 9 H-circuit different switching configuration

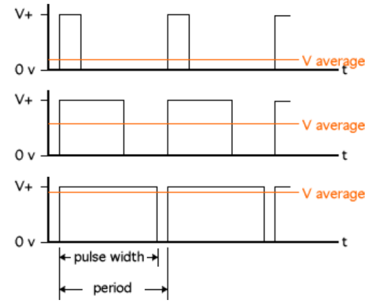

Fig. 10 Digital switching wave form

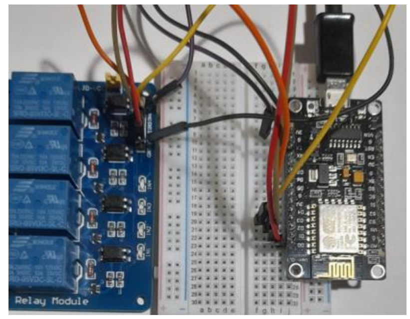

Fig. 11 Wring Connection of four channel SSR circuit with ESP8266

C. *Algorithm of Time Difference of Arrival (TDoA) Using LabVIEW*

The TDOA method is based on the evaluation of time difference measurements. All the implemented severs are in charge of assigning a timestamp to every received sample from the target. The differences between the time of arrival of the samples at each station are evaluated without any information on the effective sample transmission time. The system involving all the time difference measurements. The majority of these methods are iterative algorithms. A direct method based on a linear least squares (LLS) problem can be deduced from the signal propagation time equations. This method is implemented using LabVIEW to provide a conservative estimation of the network accuracy. The acquired measurements, the difference of reception times between a pair of stations, can be directly related to the distance of each station from the target

$$-2\begin{bmatrix} (x_2-x_1) & (y_2-y_1) & (z_2-z_1) & c\Delta t_{12} \\ & \vdots & & \\ (x_M-x_1) & (y_M-y_1) & (z_M-z_1) & c\Delta t_{1M} \end{bmatrix} \begin{bmatrix} x_T \\ y_T \\ z_T \\ d_1 \end{bmatrix} =$$

$$\begin{bmatrix} c^2\Delta t_{12}{}^2 - x_2{}^2 - y_2{}^2 - z_2{}^2 + x_1{}^2 + y_1{}^2 + z_1{}^2 \\ c^2\Delta t_{13}{}^2 - x_3{}^2 - y_3{}^2 - z_3{}^2 + x_1{}^2 + y_1{}^2 + z_1{}^2 \\ \vdots \\ c^2\Delta t_{1M}{}^2 - x_M{}^2 - y_M{}^2 - z_M{}^2 + x_1{}^2 + y_1{}^2 + z_1{}^2 \end{bmatrix} \quad (8)$$

where d_i is the distance of the i-th station from the target, D_{t1i} is the measured time difference between the arrival at the first (reference) station and the i-th station, c is the speed of light, $[x_T, y_T, z_T]$ are the coordinates of the target and $[x_i, y_i, z_i]$ are the coordinates of the i-th station.

$$Ax = b \quad (9)$$

by renaming the matrices and vectors composing the LLS estimation of the TDOA position as in (9), the estimation of the target position is then found through the following:

$$\hat{x} = (A^T A)^{-1} A^T b \quad (10)$$

where $\hat{x}$ is the estimated position through the TDOA method.

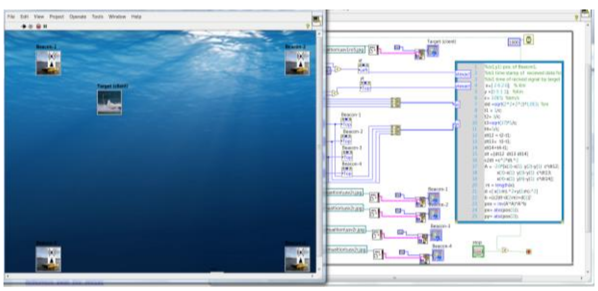

Fig. 12 LabView Program of TDoA implementation

VI. CONCLUSION

A complete survey of wireless sensors of distance estimation between sensors nodes is introduced. The configuration of A Wi-Fi communication (HTTP) between two ESP8266 NodeMCU boards is configured to exchange data without the need to connect to the internet. In position control of USV is simply represented via switching control of H-Bridge circuit. Finally a simulated algorithm based on TDoA using LabVIEW is implemented. Accuracy of calculation depends on the quantity of points of access simultaneously available to the target.

REFERENCES

[1] P. Rempel, A. Borisov, M.C.I., XVIII I.S.T.C.32, (2017)

[2] Ladd, A. M., Bekris, K. E., Rudys, A. P., Wallach, D. S., & Kavraki, L. E. (2004, June). On the Feasibility of Using Wireless Ethernet for Indoor Localization. IEEE Transactions on Robotics and Automations, 20(3), 555-559, IEEE Robotics and Automation Society.

[3] Patwari, N., Ash, J. N., Kyperountas, S., Hero III, A. O., Moses, R. L., & Correal, N. S. (2005, July). Locating the Nodes: Cooperative Localization in Wireless Sensor Networks. IEEE Signal Processing Magazine, Vol. 22, Issue 4, pp. 54-69, IEEE Signal Processing Society.

[4] [Bulusu, N., Estrin, D., Girod, L., & Heidemann, J. (2001, July). Scalable Coordination for Wireless Sensor Networks: Self-Configuring Localization Systems, In Proceedings of the Sixth International Symposium on Communication Theory and Applications, Ambleside, Lake District, UK.

[5] Patwari, N., Hero III, A. O., Perkins, M., Correal, N. S., & O'Dea, R. J. (2003, July). Relative Location Estimation in Wireless Sensor Networks. IEEE Transactions on Signal Processing, 51(8), 2137-2148. IEEE Signal Processing Society.

[6] Priyantha, N. B. (2005). The Cricket Indoor Location System. Doctoral Dissertation, Massachusetts Institute of Technology, Cambridge, USA.

[7] Priyantha, N. B., Chakraborty, A., & Balakrishnan, H. (2000). The Cricket Location-Support System, In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Mobicom '00.

[8] Priyantha, N. B., Miu, A. K., Balakrishnan, H., & Teller, S. (2001, July). The Cricket Compass for Context-Aware Mobile Applications. In Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, Mobicom '01.

[9] P. Rempel, A. Borisov, M.C.I., XVIII I.S.T.C.32, (2017)

[10] I.Averin, A. Semenov., R.R.C., IV All-R. C. 475, (2010)

[11] F. Alsehly, R. Mohd Sabri, Z. Sevak, T. I.T.M. 131, (2010)