

### **Trilateration and filtering of Wi-Fi RSSI signals as sensors for an indoor positioning system.**

### **Trilateración y filtrado de señales RSSI Wi-Fi como sensores para un sistema de posicionamiento en interiores.**

Jorge Armando GARCÍA-BAUTISTA<sup>1\*</sup>, German CUAYA-SIMBRO<sup>2</sup>

<sup>1</sup>*Instituto Tecnológico Superior del Occidente del Estado de Hidalgo, ITSOEH. Paseo del Agrarismo 2000. Carr. Mixquiahuala - Tula, km 2.5 Mixquiahuala de Juárez, Hidalgo.*

<sup>2</sup>*Instituto Tecnológico Superior del Oriente del Estado de Hidalgo, ITESA. Carretera Apan-Tepeapulco Km 3.5, Colonia Las Peñitas, Apan Hidalgo, México.*

Sent date: 12/May/2022 Acceptance date: 1/July/2022

#### **Abstract:**

This work investigates the feasibility of using the RSSI signal emitted by Wi-Fi devices (Beacons) as sensors for locating items indoors. Although there are reported works in the literature that use RSSI signals, they are complex and costly due to the high interference that this type of signal presents and the need to combine different techniques to minimize interference. The technique proposed in this work presents the application of a signal filtering method, the Kalman filter, which allows improving the readings of the RSSI signals of the devices used in the tests. To reduce the complexity of the system, the trilateration location method is used, which uses the estimation of the distances between three fixed devices, called reference nodes, to a scanning device, called the target node, which is the element to locate; using for this the relationship between distance and the RSSI signal emitted by Wi-Fi devices. Finally, a sampling of the tests carried out is presented and the effectiveness of determining the location of an object indoors through the use of the proposed system is observed, with the main conclusion of this work being that the proposed system with economical hardware, allows you to make practical rough estimates of which section or room the item is in.

**Keywords:** Location estimation, Kalman filter, RSSI, Beacons.

---

#### **Resumen:**

Este trabajo investiga la factibilidad del uso de la señal RSSI emitida por dispositivos Wi-Fi (Beacons), como sensores para la localización de elementos en interiores. Aunque existen en la literatura trabajos reportados que emplean señales RSSI, éstos son complejos y costosos debido a la alta interferencia que presentan este tipo de señales y la necesidad de combinar diferentes técnicas para minimizar las interferencias. La técnica propuesta en este trabajo presenta la aplicación de un método de filtrado de señales, filtro de Kalman, el cual permite mejorar las lecturas de las señales RSSI de los dispositivos utilizados en las pruebas. Para reducir la complejidad del sistema se emplea el método de localización por trilateración, la cual utiliza la estimación de las distancias entre tres dispositivos fijos, llamados nodos de referencia, hasta un dispositivo de escaneo, llamado nodo objetivo el cual es el elemento a localizar; utilizando para ello la relación entre distancia y la señal RSSI emitida por los dispositivos Wi-Fi. Finalmente se presenta un muestreo de las pruebas realizadas y se observa la efectividad de la determinación de la ubicación de un objeto en interiores a través del

uso del sistema propuesto, teniendo como conclusión principal de este trabajo, el que el sistema propuesto con hardware económico permite realizar estimaciones prácticas aproximadas sobre en cual sección o habitación se encuentra el elemento.

**Palabras clave:** Estimación de ubicación, Filtro Kalman, RSSI, Beacons.

---

\* *Corresponding author. E-mail: jgarcia@itsoeh.edu.mx*  
Tel. 773-179-83-74

## 1. Introducción

En la actualidad está muy extendido el uso de aplicaciones de mapas para dispositivos móviles como ayuda para viajar, especialmente en automóvil. Estas aplicaciones suministran la ubicación actual en tiempo real con una precisión de entre 5 y 10 metros, y pueden calcular la ruta más óptima para trasladarse de una ubicación a otra. Para ello, la aplicación hace el uso del sistema de satélites de posicionamiento global GPS (Global Positioning System) (Windarni et al., 2016).

El GPS funciona bien para entornos al aire libre, sin embargo, existe la demanda de sistemas robóticos capaces de navegar autónoma o semi-autónomamente en interiores para realizar actividades que apoyen en la realización de alguna tarea, por ejemplo, se podría utilizar para ubicación en supermercados y almacenes, en respuestas de emergencias de incendios, en visitas guiadas, vigilancia, etc. Precisamente en interiores, el desempeño del GPS es ineficiente. Esto debido a básicamente dos factores: el primer factor es que las señales de GPS son muy débiles en interiores; y el segundo factor es la precisión, ya que, en interiores, una precisión de ubicación con un error de 5 a 10 metros no es muy útil. Por lo que se hace evidente la necesidad de la investigación de técnicas de posicionamiento en interiores que el GPS no puede cubrir.

Se han realizado diferentes propuestas de posicionamiento en interiores por parte universidades e institutos de investigación, e incluso empresas multinacionales como Apple, o Google, han incursionado al mercado de servicios basados en la ubicación en interiores, Apple con sus dispositivos ibeacons y Google con su Eddystone (Fard et al., 2015). La mayoría de las propuestas de posicionamiento en interiores se basan en tecnologías de radio frecuencia como, por ejemplo, BLE (Bluetooth Low Energy), UWB (ultra-wideband) y RFID (Radio Frequency Identification). Estos servicios son caros de implementar, pues requieren instalación de hardware adicional y el costo del servicio de localización (Nicholas et al., 2020).

Con estos antecedentes, este trabajo de investigación, presenta resultados de la factibilidad del desarrollo de un sistema de posicionamiento en interiores económico, el cual se puede pensar como un análogo de un sistema GPS en dos dimensiones, utilizando para ello, las

tarjetas de desarrollo NodeMCU que pueden funcionar como emisores o receptores de señales Wi-Fi y que tienen como ventaja adicional el ser muy económicas si se comparan con el hardware de sistemas de posicionamiento en interiores comerciales.

## 2. Materiales y Métodos

A continuación, se presenta una revisión de diferentes técnicas de posicionamiento, así como del hardware que da soporte a este trabajo de investigación.

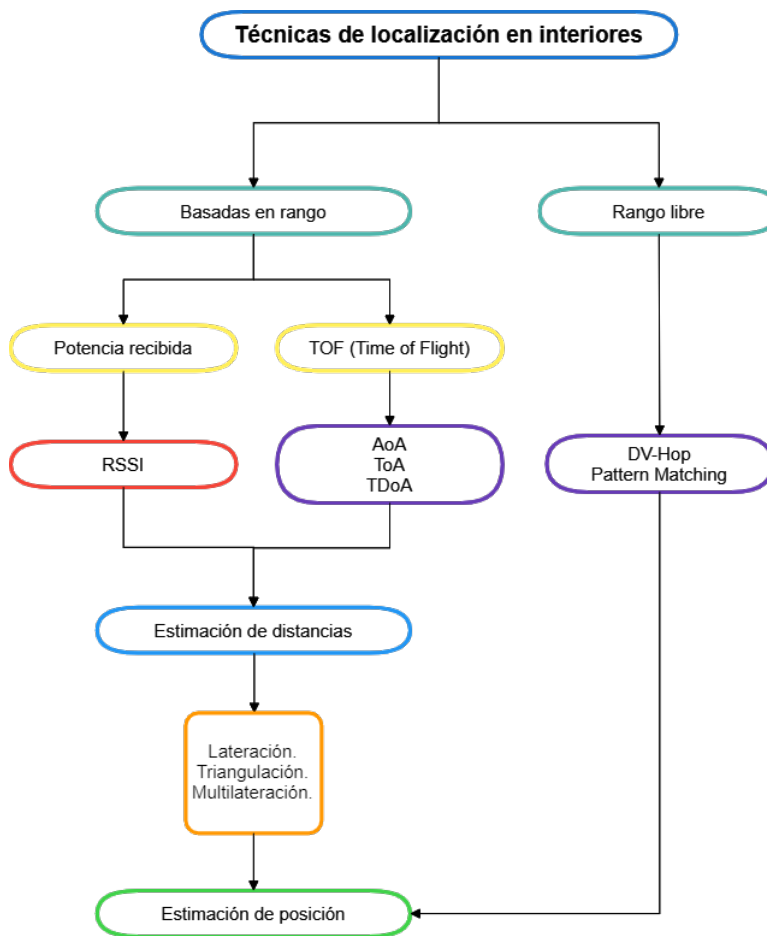
Se han propuesto diferentes técnicas de posicionamiento en interiores basadas en radio frecuencia (RF). Entre ellas están:

- *Trilateración*. Estima la posición en base en las distancias del objeto a localizar a los nodos de referencia.
- *Tecnologías basadas en Filtros*. Basada en búsqueda Montecarlo, utiliza distribuciones de probabilidad para estimar iterativamente la posición y orientación del objeto a localizar.
- *Fingerprinting*. Se recolectan y almacenan apriori las características de las señales de RF para varios ángulos en diferentes posiciones en el entorno y se comparan con las actuales para estimar la posición.
- *Basado en celdas*. Similar al Fingerprinting, se divide el entorno en áreas y se obtienen offline las características de las señales RF. Su diferencia radica en que utiliza los parámetros de la señal para romper un empate en la localización.
- *Triangulación*. Similar a la Trilateración, determina la posición de un objeto o un nodo móvil usando no la distancia, sino los ángulos de llegada de las señales inalámbricas.

De manera general, todas estas técnicas de posicionamiento están basadas en la obtención de datos de las señales RF utilizadas y similitudes en sus características o en la estimación de distancias con el objeto móvil a localizar. El sistema de localización propuesto en este trabajo utiliza la técnica trilateración de señales RSSI. Esto es debido al equilibrio entre la economía y la facilidad de implementación del sistema, contra la pérdida de precisión para cálculo de las coordenadas. Precisión que se mejora agregando el filtro de Kalman a los datos de las señales RSSI obtenidas. Las operaciones del filtro de Kalman son muy sencillas, por lo que el sistema se podría utilizar en aplicaciones en tiempo real.

### 2.1. Clasificación de las técnicas de localización en interiores

Se puede establecer una clasificación simplificada de las técnicas de posicionamiento en interiores, como se muestra en la figura 1.



**Figura 1** Clasificación simplificada de las técnicas de posicionamiento en interiores.

Como se observa, se pueden diferenciar las técnicas de posicionamiento en interiores en dos grupos principales:

- *Basado en rango.* Utiliza parámetros de la señal para estimar distancias e incluye posicionamiento basado en tiempo de llegada (TOA), posicionamiento basado en diferencia de tiempo de llegada (TDOA), posicionamiento basado en ángulo de llegada (AOA) y posicionamiento basado en el indicador de intensidad de la señal recibida (RSSI).
- *Rango libre.* Estas se basan en las coincidencias de los parámetros entre las señales. Primero se recopilan una serie de posiciones de referencia en el entorno y se crea una base de datos de parámetros de posición y características, luego los parámetros medidos se comparan con la base de datos para obtener la información de la posición objetivo actual.

De manera particular, este trabajo se encuentra en la categoría de técnicas basadas en rango, las cuales tienen las siguientes ventajas y desventajas:

Basadas en potencia recibida (RSSI):

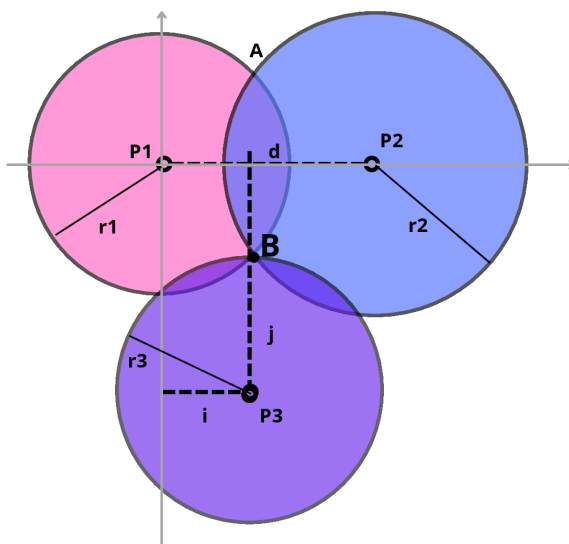
- Baja complejidad en la implementación.
- Sin tiempos de sincronización.
- Bajo consumo de energía.
- Baja precisión.

Basadas en TOF (ToA, TDoA, AoA):

- Alta complejidad en la implementación.
- Necesitan tiempos de sincronización.
- Alto consumo de energía.
- Alta precisión.

## 2.2. Trilateración

El sistema propuesto en este trabajo utiliza la técnica de Trilateración para estimar la posición. Esta técnica requiere conocer las coordenadas de tres elementos llamados nodos de referencia, los cuales tienen una posición fija y conocida  $(x,y)$  dentro del entorno. Si se conoce además la distancia desde el elemento móvil a localizar a cada uno de los tres nodos de referencia, es posible calcular la posición  $(x,y)$  de este elemento móvil dentro de dicho entorno, como se puede observar en la figura 2.



**Figura 2** Trilateración 2D. Se requieren 3 nodos de referencia que se encuentran en los puntos P1, P2 y P3, con coordenadas conocidas y fijas, para poder estimar la posición del cuarto nodo móvil B. Fuente:

Elaboración propia (Adaptado de Wikipedia).

En otras palabras, si conocen, tanto las coordenadas de tres puntos de referencia, como las distancias desde estos puntos de referencia hasta el objeto a localizar, se puede encontrar las coordenadas 2D de este objeto dentro del inmueble.

Las coordenadas (x,y) del elemento B a localizar (Figura 2), utilizando la técnica trilateración, están dadas por:

$$x = \frac{r_1^2 - r_2^2 + d^2}{2d} \quad (1)$$

$$y = \frac{r_1^2 - r_3^2 + i^2 + j^2}{2j} - \frac{i}{j}x \quad (2)$$

Donde:

- $x$ : coordenada horizontal del nodo móvil B (m).
- $y$ : coordenada vertical del nodo móvil B (m).
- $r_1$ : distancia del nodo móvil B al nodo de referencia P1 (m).
- $r_2$ : distancia del nodo móvil B al nodo de referencia P2 (m).
- $r_3$ : distancia del nodo móvil B al nodo de referencia P3 (m).
- $d$ : distancia entre los nodos de referencia P1 y P2 (m).
- $j$ : distancia del segmento de recta que pasa por P3 y B a la recta d (m).
- $i$ : distancia de la intersección de j con d a P1 (m).

Dado que las posiciones de los nodos de referencia son fijas y conocidas de antemano, entonces para usar la trilateración, bastaría encontrar una forma de calcular las distancias del elemento a localizar a cada uno de los tres nodos de referencia. Para ello se podría utilizar la medición del indicador de fuerza de la señal recibida (RSSI), ya que existe una relación entre la intensidad de esa señal y la distancia entre emisor y receptor.

La RSSI es una escala de referencia (en relación con  $1mW$ ) para medir el nivel de potencia de las señales recibidas dispositivos conectados a redes inalámbricas. Cuanto mayor sea la distancia entre el emisor y el receptor de una señal, menor será la intensidad de la señal RSSI y viceversa. Esta variable se puede obtener de dispositivos que emitan señales Bluetooth o señales WiFi, a los cuales se les nombra en este documento indistintamente balizas, nodos o Beacons.

La relación entre la señal RSSI y la distancia sigue a las ecuaciones matemáticas:

$$RSSI = -(10n \log_{10} d + A) \quad (3)$$

$$d = 10^{\left(\frac{A-RSSI}{10*n}\right)} \quad (4)$$

Donde:

- $A$ : Valor RSSI promedio medido a la distancia de un metro al nodo (dBm).
- $n$ : constante de propagación que depende del entorno.
- $d$ : distancia al nodo (m)

Estimando  $A$  como el promedio de varias lecturas, junto con la lectura del RSSI a distancias predefinidas  $d$ , se puede resolver la constante de propagación  $n$ . Debido a las fluctuaciones en los valores de RSSI, el promedio de varias lecturas de RSSI en cada distancia se sustituye en la ecuación para obtener la constante de propagación  $n$ . En el caso de que se obtengan diferentes valores de  $n$  para cada distancia,  $n$  se calcula como la media de todos los  $n$  valores obtenidos (Dahlgren et al, 2014).

### 2.3. Filtro de Kalman

El filtro de Kalman es un algoritmo matemático recursivo, que permite calcular un estimador lineal óptimo del estado actual de un proceso dinámico. Su importancia radica en que permite filtrar datos de sistemas variantes en el tiempo y que se puedan expresar como una tupla estado-espacio. Este filtro presupone que todas las señales son inherentemente ruidosas, por lo que a su entrada permite el ingreso de una serie de datos en "crudo", es decir, con ruido, los procesa junto con el histórico de datos y devuelve una estimación más precisa del estado "real" actual. Esta estimación se calcula realizando predicciones sobre estado actual y sus respectivas correcciones basados en la covarianza del error.

El algoritmo del filtro de Kalman se realiza esencialmente dos etapas: predicción y corrección. En la etapa de predicción se realiza una estimación del estado actual (tiempo  $t$ ) en base en el estado anterior (tiempo  $t - 1$ ) y la covarianza de los errores de predicción. En la etapa de corrección se intenta minimizar los errores obtenidos en la etapa de predicción con la ganancia  $K$ . En ambas etapas las operaciones implicadas son sencillas y rápidas de calcular por lo que son ideales para funcionar en procesos en tiempo real. El algoritmo del filtro de Kalman se muestra a continuación:

---

### Algoritmo 1 Filtro simple de Kalman

---

**Entrada:**  $z_t, R, Q$

$z_t$ : observación del estado actual  
 $R$ : covarianza del ruido intrínseco  
 $Q$ : covarianza del ruido observado

**Salida:**  $x, P_t$

*Predicciones:*

$$\bar{x}_t = Ax + Bu$$

$$\bar{P}_t = AP_tA + R$$

*Correcciones:*

$$K = \frac{\bar{P}_t C}{C \bar{P}_t C + Q}$$

$$x = \bar{x}_t + K(z_t - C \bar{x}_t)$$

$$P_t = \bar{P}_t - KC \bar{P}_t$$


---

#### 2.4. Hardware utilizado (Beacons)

Tanto los nodos de referencia, como el nodo móvil a localizar, con los cuales se realizaron las pruebas de localización en interiores, son tarjetas de desarrollo NodeMCU (véase la figura 3), las cuales utilizan el procesador ESP8266 y son capaces de conectarse a redes WiFi y además pueden configurarse como Access Point o como escaner Wi-Fi para detectar redes cercanas. Otra ventaja es que resulta muy sencillo obtener los parámetros de la señal RSSI, ya que el firmware se puede programar con Arduino IDE.



**Figura 3** Tarjeta de desarrollo NodeMCU. Fuente: Amica

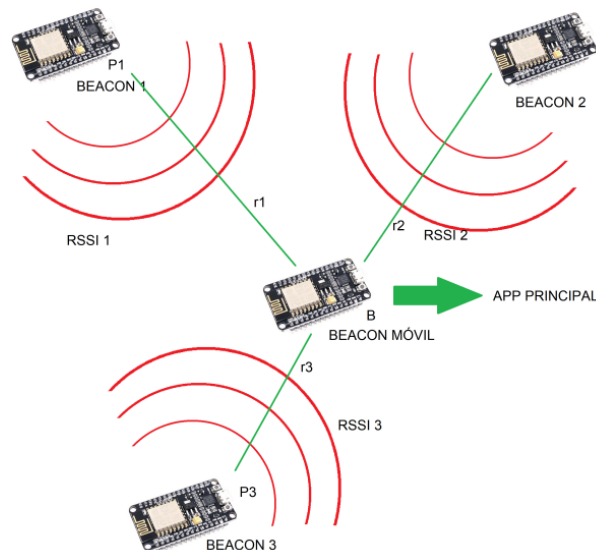
El sistema de posicionamiento implementado en este trabajo se clasifica dentro de las técnicas basadas en rango, específicamente en la obtención de la potencia de la señal RSSI para el cálculo de la distancia entre nodos e implementa la técnica de Trilateración para la estimación de la posición. También incluye un filtrado de la señal RSSI y un filtrado de las distancias calculadas para intentar minimizar el ruido inherente a este tipo de sistemas haciendo uso del filtro simple de Kalman.

### 3. Resultados y Discusión

#### 3.1. Sistema de localización propuesto

El sistema implementa el algoritmo de trilateración, el cual necesita tres Beacons de referencia P1, P2 y P3. Como las señales RSSI son fácilmente reflejadas por superficies sólidas debido a su baja energía, los Beacons se colocaron en una superficie rectangular de 1.4x 1.4m, a la altura del piso para intentar minimizar las reflexiones de las señales RSSI con otros objetos o con el mismo suelo.

Las posiciones fijas de los Beacons de referencia en la superficie rectangular de 1.4x 1.4m para pruebas, iniciando desde la esquina inferior izquierda dadas en metros son: P1(0, 1.05), P2 (1, 0.7), P3(1.05, 0). El beacon móvil a localizar B se coloca en una posición inicial arbitraria dentro del entorno ya que se irá moviendo para poder realizar las estimaciones de su posición. (véase la figura 4).



**Figura 4** Esquema simplificado de la configuración del sistema de localización por Trilateración. Tres NodeMCU funcionan como Beacons de referencia y un NodeMCU como Beacon móvil a localizar.

### 3.2. Obtención y envío de datos RSSI a la App principal

Los datos son recibidos primero por el cuarto Beacon B que hace las veces de objeto a localizar.

El firmware de este cuarto Beacon detecta las señales WiFi-cercanas y discrimina entre ellas para obtener solo los datos de las SSID de los 3 Beacon NodeMCU configurados en modo Access Point, guarda los valores individuales RSSI de cada nodo y los envía en una cadena de caracteres por medio de comunicación serial RS232 (Figura 5).

```

SignalStrengthWiFi
#include "ESP8266WiFi.h"
const int RSSI_MAX = -50; //
const int RSSI_MIN = -100; //

int count = 0;

int scanDelayTime = 100;
int WiFiDisplayTime = 1000;
long A;
float nssi = 3;
float distance;

void setup() {
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  Serial.begin(115200);
  delay(1500);
  Serial.println("-----");
  Serial.println("Armando Nod");
  Serial.println("Wifi Scan c");
  Serial.println("-----");
  delay(2000);
}
  
```

```

20:57:49.335 -> 1) NodeMCU 1
20:57:51.507 -> Señal: -33%. Distancia: 12.59
20:57:51.507 -> Scan finalizado
20:57:51.507 -> -----
20:57:51.507 -> 7 Access Point encontrado
20:57:51.507 -> -----
20:57:51.507 -> 1) NodeMCU 1
20:57:51.507 -> Señal: -32%. Distancia: 11.66
20:57:53.683 -> Scan finalizado
20:57:53.683 -> -----
20:57:53.683 -> 5 Access Point encontrado
20:57:53.716 -> -----
20:57:53.716 -> 1) NodeMCU 1
20:57:53.716 -> Señal: -33%. Distancia: 12.59
20:57:55.884 -> Scan finalizado
20:57:55.884 -> -----
20:57:55.884 -> 4 Access Point encontrado
20:57:55.884 -> -----
20:57:55.884 -> 1) NodeMCU 1
20:57:55.884 -> Señal: -35%. Distancia: 14.68
  
```

**Figura 5** Obtención de y envío de datos RSSI desde el Beacon móvil a la App .NET. Fuente: Elaboración propia.

Para la obtención de datos y estimación de la posición, se desarrolló una aplicación de escritorio para el sistema operativo Windows en la plataforma .NET, la cual acepta los datos de la medición de RSSI enviados por el Beacon móvil por medio de comunicación serial RS232. Estos datos RSSI se utilizan en la ecuación (4), para calcular las tres distancias  $r_1$ ,  $r_2$  y  $r_3$ , las cuales son necesarias para estimar la posición del Beacon Móvil (véase la figura 6).

```

if (nBeacon3 == 1)
{
  Beacon1Data = Convert.ToDouble(datos[5]);
}
else if (nBeacon3 == 2)
{
  Beacon2Data = Convert.ToDouble(datos[5]);
}
else if (nBeacon3 == 3)
{
  Beacon3Data = Convert.ToDouble(datos[5]);
}

double r1 = Math.Pow(10.0, (double)(Ao1 + kalman1.filter(Beacon1Data)) / 20.0);
double r2 = Math.Pow(10.0, (double)(Ao2 + kalman2.filter(Beacon2Data)) / 20.0);
double r3 = Math.Pow(10.0, (double)(Ao3 + kalman3.filter(Beacon3Data)) / 20.0);
  
```

**Figura 6** Código que ordena los vectores de datos RSSI obtenidos de los Beacons de referencia e implementa el cálculo de las distancias  $r_1$ ,  $r_2$  y  $r_3$ , las cuales son necesarias para la trilateración. Fuente: Elaboración propia.

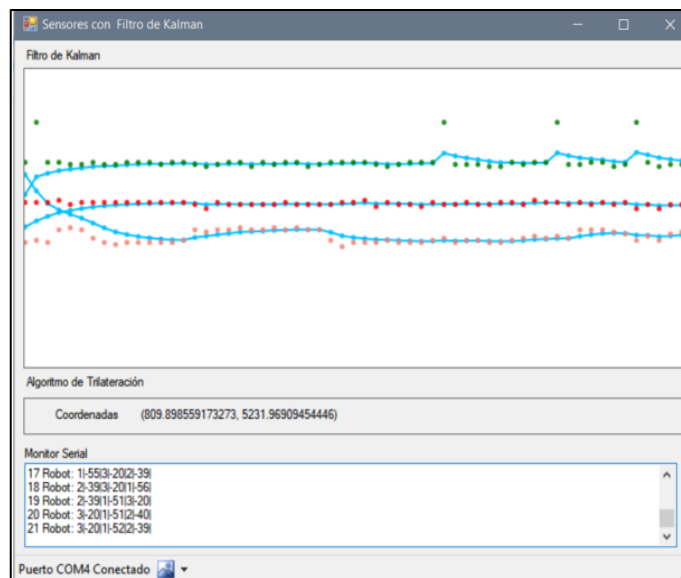
### 3.3. Filtrado de señales RSSI

Cuando se utiliza el valor RSSI para realizar el cálculo distancias entre Beacons, que hacen las veces de nodos de referencia, se encuentra con el problema de que las lecturas son muy inconsistentes, es decir, en ocasiones las medidas obtenidas del RSSI varían de manera muy abrupta obteniéndose valores atípicos.

Esto es así, debido principalmente que la RSSI es muy susceptible a las reflexiones con elementos contenidos dentro del espacio de medición, e incluso con las mismas paredes del espacio en cuestión. Por lo que, algunos autores desaconsejan el uso de la RSSI como estimador de distancias por estas problemáticas (Dong, Q. y Dargie, W., 2012).

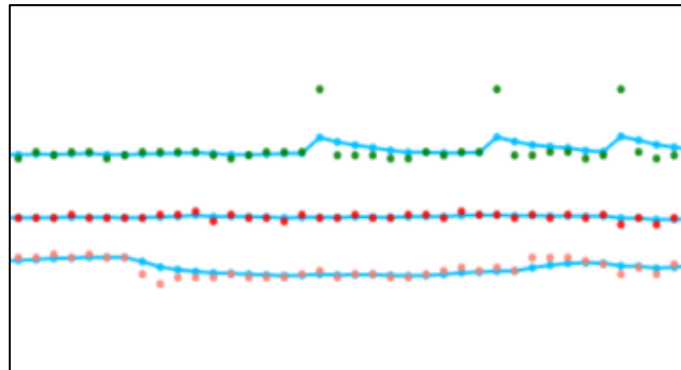
Por ello, si se utilizan los “datos crudos” de RSSI, habrá discrepancias muy grandes entre los cálculos para una misma distancia. Ya que la medida de RSSI varía de manera muy significativa, debido a los problemas de reflexión mencionados antes. Por ello se debe implementar alguna técnica de filtrado para que los datos obtenidos sean más consistentes con la realidad. El filtro que se utilizó en el sistema de posicionamiento es el filtro simple de Kalman.

En este caso se desea conocer un valor de RSSI más cercano a la realidad en base a las muestras obtenidas, que se asumen ruidosas, y con ello poder realizar una estimación más adecuada de las distancias entre Beacons. Para ello, la aplicación selecciona los datos RSSI provenientes del Beacon móvil y los guarda en tres vectores de datos y los pasa al algoritmo de filtrado de Kalman, el cual es una clase desarrollada en .NET (Figura 7).



**Figura 7** Aplicación desarrollada en .NET. Se aplica el filtro de Kalman (líneas azules) a los tres conjuntos datos RSSI (puntos verdes, rojos y naranjas).

Al pasar cada uno de los tres vectores de datos por el filtro de Kalman se observa que se obtiene un vector de salida con datos más consistentes y robustos eliminando parte ruido ambiente obtenido por los datos RSSI “en bruto”. Tal como se muestra en la figura 8.



**Figura 8** La señal filtrada (azul) obtiene datos más consistentes que los datos originales, incluso para datos atípicos, lo cual es más notable para la señal original verde.

### 3.4. Cálculo de la posición por trilateración

Una vez realizado el filtro de los datos de las señales RSSI, la información es utilizada para calcular las distancias entre los tres Beacons de referencia basados en la placa NodeMCU y el cuarto Beacon que hace las veces de objeto a localizar. En la figura 9 se muestra un extracto de los valores RSSI filtrados con su correspondiente distancia.

```

170 | | | | | {
171 | | | | |     plt.clear();
172 | | | | |     double[] node1 = filtrarRSSI(testData);
173 | | | | |     System.Diagnostics.Trace.WriteLine("\n\n");

```

---

Salida

Mostrar salida de: Depurar

```

RSSI: -55.0158443019116 D = 2.50730856889552
RSSI: -54.2749806246388 D = 2.730552518974
RSSI: -53.7161153973372 D = 2.91201917332334
RSSI: -53.5401965389642 D = 2.97159879146897
RSSI: -53.1618362665667 D = 3.10390332876302
RSSI: -52.8764228954945 D = 3.20759003233336
RSSI: -52.6611233560619 D = 3.28809102867819

```

---

BEACON 2

```

RSSI: -49.3051709022719 D = 1.53017624270297
RSSI: -46.2823178463803 D = 2.16712572597818
RSSI: -44.2477073553408 D = 2.73914253681709
RSSI: -42.9585705898702 D = 3.17739692206063
RSSI: -41.7404623344476 D = 3.65575332170897
RSSI: -40.8215910657808 D = 4.06368884419221
RSSI: -39.8827899452347 D = 4.52752130607248
RSSI: -39.1746120203075 D = 4.9121248769079
RSSI: -38.8860601941465 D = 5.07805020509457
RSSI: -38.6683931835172 D = 5.20691323405281
RSSI: -38.5041976275693 D = 5.30627945831131
RSSI: -38.1346807446853 D = 5.53689085658119
RSSI: -37.8559383213761 D = 5.71745932879012
RSSI: -37.8913201009187 D = 5.69421151115859
RSSI: -37.9180241285347 D = 5.67673725486665
RSSI: -38.429476360717 D = 5.35212418721818
RSSI: -38.3239724274732 D = 5.41753066971967
RSSI: -38.2443862884273 D = 5.46739794975601
RSSI: -38.1843510524556 D = 5.50531847515626

```

**Figura 9** Extracto de resultados del cálculo de distancias entre Beacons.

Finalmente, las ecuaciones (1) y (2) se utilizan para calcular las coordenadas (x, y) del Beacon móvil, las cuales indican una estimación de su localización actual. Para el caso inicial se tiene:

- $r1 = 0.25m$
- $r2 = 0.5m$
- $r3 = 1.25m$
- $P1 = (0, 1.05)$
- $P2 = (1, 0.7)$
- $P3 = (1.05, 0)$

La posición del Beacon móvil es:

- $x = 0.538m$ ,
- $y = 1.252m$

En la figura 10 se muestra un fragmento de la implementación en código de la trilateración que realiza el cálculo mostrado.

```
double r1 = Math.Pow(10.0, (double)(Ao1 + kalman1.filter(Beacon1Data)) / 20.0);
double r2 = Math.Pow(10.0, (double)(Ao2 + kalman2.filter(Beacon2Data)) / 20.0);
double r3 = Math.Pow(10.0, (double)(Ao3 + kalman3.filter(Beacon3Data)) / 20.0);

Trilateration tri = new Trilateration();
Coordinates loc = tri.locateMe(
    0, 1.35, r1,
    1, 0.7, r2,
    0.7, 0, r3
);
lblCoord.Text = "(" + loc.x + ", " + loc.y + "');
```

**Figura 10** Fragmento de código que implementa la trilateración. Fuente: Elaboración propia

Este proceso se realizó por cuatro iteraciones de diez mediciones, cambiando en cada medición la posición del Beacon a localizar, para observar si las coordenadas calculadas por el sistema coincidían con las reales (Tabla 1).

Como se observa, el sistema de ubicación en interiores implementado permite dar un estimado de la posición, con un error promedio de 41.3 cm en el eje x y 35.8 en el eje y para esta iteración. Finalmente, se promediaron los errores de las cuatro iteraciones tanto para el eje x, como para el eje y, para obtener el error promedio en cada eje (Tabla II).

**Tabla 1.** Extracto de resultados de Trilateración

<b>Ite. 2 Resultados de la localización</b>						
<b>Medición</b>	<b>Posición</b>		<b>Posición obtenida por Trilateración</b>		<b>Errores</b>	
	x	y	x	y	ex	ey
	(cm)	(cm)	(cm)	(cm)	(cm)	(cm)
1	49	50	16	15	33	35
2	66	73	26	38	40	35
3	66	144	41	64	25	80
4	40	17	20	18	20	1
5	85	30	5	20	80	10
6	76	128	31	48	45	80
7	68	79	48	44	20	35
8	115	11	45	9	70	2
9	61	50	6	20	55	30
10	61	111	36	61	25	50
<b>Promedios:</b>					<b>41.3</b>	<b>35.8</b>

**Tabla 2.** Extracto de errores promedio de Trilateración

<b>Errores Promedio</b>					
<b>Coordenada</b>	<b>Iteración 1</b>	<b>Iteración 2</b>	<b>Iteración 3</b>	<b>Iteración 4</b>	<b>Error</b>
x	47.8	41.3	56.7	48.8	<b>48.6</b>
y	46.6	35.8	46.9	50.5	<b>51.5</b>

## 4. Conclusiones

Al usar un filtro de Kalman, se puede eliminar parte del ruido de una señal muy ruidosa. Como las funciones de actualización son fáciles de calcular, la complejidad de tiempo del filtro es muy baja; esto da como resultado un sistema de alto rendimiento y que puede ser optimizado para implementarlo en aplicaciones en tiempo real como lo es la ubicación de un robot en interiores, por ejemplo. El único inconveniente es que se pierde un poco de capacidad de respuesta, ya que la capacidad de respuesta depende del valor de la ganancia de Kalman. Si es baja, el filtro da mediciones más consistentes con las predicciones del modelo, y si es alta le da mayor peso a las mediciones más recientes, sin embargo, al final se obtiene una señal más consistente. A pesar de que los datos filtrados no dan una medida de distancia muy precisa, se puede obtener una posición aproximada para conocer, por ejemplo, en que habitación o sección se encuentra el objeto a localizar. Además, en contraste con un GPS, con el sistema propuesto podemos tener un mayor grado de exactitud.

Como trabajo futuro se realizará una aproximación de los datos de distancia vs RSSI con una técnica de ajuste de curvas, como, por ejemplo, regresión polinomial; y comparar si se obtienen mejores resultados a una precisión de decímetros.

También se explorará el método de posicionamiento RSSI basado en tiempo de vuelo (ToF) utilizando tarjetas de ultra ancho de banda (UWB) como la Deca1000, el cual se reporta como un dispositivo de mayor precisión ya que no tiene los problemas de reflexión y pérdida que se encuentran con RSSI.

## 5. Referencias

- Bulten, W., Rossum, A. C. V. and Haselager, W. F. G. (2016). Human SLAM, Indoor Localisation of Devices and Users. *IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*, 211-222. <https://doi.org/10.1109/IoTDI.2015.19>
- Bulten, W. (2015). Kalman filters explained: Removing noise from RSSI signals. Available at: [www.wouterbulten.nl/blog/tech/kalman-filters-explained-removing-noise-from-rssi-signals](http://www.wouterbulten.nl/blog/tech/kalman-filters-explained-removing-noise-from-rssi-signals). Accessed: September 24, 2021.
- Clar, M., Ramos, R. and Suriñach, J. (1998). Latent Variable Model to Measure Regional Manufacturing Production in Spain. Workshop on Regional Economic Indicators. Braga, Portugal: *University of Minho, Workshop on Regional Economic Indicators*.

- Dahlgren, E. y Mahmood, H. (2014). Evaluation of Indoor Positioning Based on Bluetooth Smart Technology. Master Thesis, Chalmers University of Technology, Goteborg, Sweden.
- Fard, H. K., Chen, Y. and Son, K. K. (2015). Indoor positioning of mobile devices with agile iBeacon deployment. *IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE '15)*, 275-279. <https://doi.org/10.1109/CCECE.2015.7129199>
- Nicholas, G., & Shapiro, A. (2021). Failed hybrids: The death and life of Bluetooth proximity marketing. *Mobile Media & Communication*, 9(3), 465–487. <https://doi.org/10.1177/2050157920975836>
- Windarni, V. A., Sedyono, E. and Setiawan, A. (2016). Using GPS and Google maps for mapping digital land certificates. *International Conference on Informatics and Computing (ICIC)*, 422-426. <https://doi.org/10.1109/IAC.2016.7905756>
- Yassin, A., Nasser, Y., Awad, M., Al-Dubai, A., Liu, R., Yuen, C., Raulefs, R. and Aboutanios, E. (2017). Recent Advances in Indoor Localization: A Survey on Theoretical Approaches and Applications. *IEEE Commun. Surv. Tutor.* 19, 1327–1346. <https://doi.org/10.1109/COMST.2016.263242>