

Geolocalización de Nodos Inalámbricos

Matías Parodi, Guillermo Reisch

Abstract—The abstract goes here.

Index Terms—Algoritmos Evolutivos, triangulación, WiFi

I. INTRODUCCIÓN

EL presente informe consiste en la evaluación del comportamiento de un algoritmo evolutivo aplicado a la triangulación de dispositivos móviles en relación a la posición de al menos tres routers inalámbricos, cuyas posiciones absolutas o relativas a un centro arbitrario sí se conocen.

Se asumirá que los routers presentes en el entorno comparten características similares (e.g. alcance máximo), lo cual resulta razonable asumir para un gran número de routers WiFi existentes en el mercado. Esto será considerado al momento de la inicialización de la población inicial.

Por otro lado, resulta inapropiado asumir que las señales no se ven afectadas por obstáculos encontrados en el entorno, lo cual obligará a estimar, además de la posición, cuál es la caída o atenuación de la señal hacia cada router y para esto se estudiarán varias alternativas de modelo de propagación de ondas.

19 Julio, 2013

II. PROBLEMA

Debido a la explosión de las redes WiFi en todo el mundo [1], en los últimos años han surgido servicios de geolocalización por triangulación WiFi [2] que ofrecen APIs [3] y SDKs [4] para el desarrollo de aplicaciones sensibles a la posición del usuario.

Desafortunadamente, ninguna de las grandes empresas en estas áreas han publicado especificaciones de sus implementaciones, por lo que se desconocen muchos detalles del funcionamiento interno y en particular de los algoritmos que utilizan.

En el ámbito científico se exploraron varias alternativas. Quizás el enfoque más sencillo consiste en la triangulación de nodos a partir simplemente de las intensidades de las señales a cada router inalámbrico, como se analiza en [5]. Este enfoque presenta varias desventajas ya que no considera muchos aspectos de la realidad como la caída de señal por ruido electromagnético u obstáculos en el ambiente, condiciones climatológicas que afectan la señal, diversidad de antenas, etc.

Otro grupo de investigadores han explorado el problema desde el área de machine learning, consiguiendo buenos resultados utilizando dos enfoques distintos. Por un lado,

se han implementado satisfactoriamente sistemas basados en *fuzzy logic* [6], mientras que otros han intentado la integración de más de un sistema de triangulación, con el fin de tomar los beneficios de cada uno de ellos, combinando los resultados a partir de constantes de peso aprendidas y que se van ajustando a medida que se ejecuta el algoritmo [7].

Dado el tamaño del espacio de búsqueda, resulta interesante atacar el problema mediante un algoritmo evolutivo con el objetivo de lograr aproximar la posición con una precisión comparable a los servicios existentes pero para eso habrá que restringir el problema de lo contrario sería imposible abordarlo dada la cantidad de variables libres presentes.

III. DIFICULTADES

El nivel de señal recibido por el dispositivo que se intenta triangular varía con el tiempo, aunque no hayan cambiados sus posiciones relativas. Es importante entender qué ocasiona dichas fluctuaciones, ya que tener una buena estimación de la señal será esencial para la correcta evaluación del individuo.

Por mayores detalles respecto a cuales son los factores más importantes que afectan a la señal recibida y como estos influyen en la misma referirse a [6] y [8].

A. Tecnología

Las cuatro características más importantes que caracterizan a los dispositivos con los que se trabaja son: la frecuencia de transmisión, el tipo de antena, la ganancia de la antena y la potencia de transmisión.

A pesar de que existen muchas marcas y modelos existentes en el mercado, los fabricantes de routers modernos respetan estándares internacionales para garantizar el correcto funcionamiento de redes WiFi con cualquier dispositivo inalámbrico.

Mediante un estudio de mercado se encontró que los routers domésticos son bastante homogéneos en cuanto a sus características, lo cual nos permite asumir algunas cosas que ayudarán a simplificar el el problema de manera de hacerlo abordable.

La frecuencia de transmisión depende del canal en el que se encuentre transmitiendo el router, sin embargo el rango en el que pueden estar transmitiendo no es demasiado grande, pudiendo variar entre 2,412GHz a 2,472GHz. Para simplificar el problema se asumirá entonces que el router opera en una

frecuencia similar al promedio, 2,442GHz, correspondiente al canal 7.

En cuanto al tipo de antena, se encontró que todos los routers modernos tienen entre una y dos antenas isotrópicas de ganancia 5dBm, que emiten uniformemente hacia todos lados, por lo que la energía se disipará en el ambiente proporcionalmente al cuadrado de la distancia.

En cuanto a la potencia de transmisión, muchos routers permiten configurarla, pudiendo variar de 1dBm (1mW) a 20dBm (100mW), sin embargo en la práctica la mayoría de routers usan la potencia máxima con el fin de hacer llegar la señal más lejos.

B. Entorno

El entorno puede afectar significativamente la calidad e intensidad de la señal, en particular el ruido electromagnético, condiciones climáticas y obstáculos presentes ocasionan una pérdida significativa de la misma.

Dado que es imposible construir un modelo de propagación para cada entorno, ya que este afecta de manera diferente a cada dispositivo, resulta vital considerar la caída de la señal percibida por el dispositivo móvil, en lugar de intentar modelar cada variable por separado.

Como se verá más adelante, cada router tendrá asociada una caída de señal, la cual el algoritmo intentará aproximar de manera de encontrar la posición más probable del individuo en relación a ellos.

Se tendrá entonces que la señal real será igual a la señal emitida por el router, menos la pérdida por disipación (dependiente de la distancia), menos la pérdida por obstáculos (dependiente del entorno).

C. Muestreo

Una tarjeta WiFi estándar solo puede escuchar un canal por vez, por lo que para obtener todas las redes visibles en un entorno es necesario iterar en todos los canales en busca del anuncio de los routers inalámbricos.

Esto ocasiona que la información obtenida sobre una red en un determinado momento esté desactualizada por algunos segundos. Esto, sumado a los factores antes mencionados, ocasiona que mediciones consecutivas pueden dar valores de intensidad de señal relativamente distintos.

Para evitar este problema se tomará como intensidad de la señal hacia un router el promedio de las cinco últimas lecturas.

D. Cantidad de Nodos

La cantidad de nodos afectará la precisión con la que se consiga triangular al dispositivo móvil pero serán necesarios al menos 3 routers para un correcto posicionamiento del mismo.

IV. MODELOS EXISTENTES

El modelo de propagación de ondas permitirá la traducción de una intensidad de señal a una distancia y viceversa.

Encontrar una buena aproximación de dt , la distancia teórica, es esencial ya que a partir de este valor será posible encontrar una aproximación del error entre la distancia real y la distancia estimada del nodo.

Existen muchas aproximaciones que fueron analizadas en [9], [8] y [10], aquí se describirán las 4 más representativas con el objetivo de analizar distintos enfoques posibles.

A. Empirical Propagation Model

Este modelo de propagación es quizás el más preciso de todos a pesar de ser muy simple. La principal desventaja, que lo hace impracticable en muchos casos, es que requiere del estudio preciso del ambiente en cuestión, pero como vimos antes esto depende fuertemente de las condiciones en un instante dado, por lo que su utilidad es limitada a medida que avanza el tiempo.

$$dt(rs) = |rs - ts| \times k$$

En donde rs es la intensidad de señal recibida, ts la intensidad de señal transmitida por el router y k es un valor empírico que corresponde a la distancia cubierta por 1dBm.

B. Wall Attenuation Factor Propagation Model

Este modelo es una adaptación del *Floor Attenuation Factor Propagation Model* sugerido en [11].

En lugar de considerar la atenuación de la señal debido al factor del piso del edificio, considera el efecto causado por las paredes entre el transmisor y receptor.

La ecuación que relaciona la potencia en función de la distancia es:

$$P(d) = P(d_0) - 10 \times n \times \log\left(\frac{d}{d_0}\right) - \begin{cases} n \times W \times WAF & nW < C \\ C \times WAF & nW \geq C \end{cases}$$

En donde n indica el ratio de incremento de la atenuación debido a la distancia, $P(d_0)$ es la potencia de la señal en la distancia de referencia d_0 , d es la distancia entre el transmisor y el emisor, C es la cantidad de paredes hasta la que el factor de atenuación se ve afectado y WAF es el factor de atenuación.

Como se puede apreciar este modelo es bastante más complejo y sigue requiriendo un análisis empírico para determinar el factor de atenuación.

C. Land Propagation Model

Un tercer modelo teórico bastante más complejo considera la pérdida de potencia en la señal debido a tres factores principales:

- *Path loss* (L_p): se refiere a la perdida promedio debido a la propagación de las ondas en el espacio
- *Slow fading* (L_s): se refiere a perdidas de señal causadas en distancias largas debido a variaciones lentas en el espacio-tiempo [12]
- *Fast fading* (L_f), se refiere a perdidas de señal por variaciones bruscas en el espacio [13] [14].

De esta forma la ecuación que relaciona potencia con distancia está dada por:

$$P_r = \frac{G_r \times G_t}{L} \times P_t$$

$$L = L_p \times L_s \times L_f$$

En donde G_r y G_t son las ganancias de las antenas de recepción y transmisión respectivamente y P_t es la potencia con la que transmite el router.

Calcular L es extremadamente complejo y requiere de mediciones del ambiente en donde se va a desenvolver el algoritmo, lo que lo hace poco práctico en este contexto.

D. Free Space Propagation Model

Este modelo de propagación está inspirado en la ecuación de propagación de Friis [15] y es quizás el más útil de todos ya que ofrece un compromiso entre simplicidad y utilidad.

$$P_r = \frac{G_t \times G_r \times \lambda^2}{(4\pi \times d)^2} \times P_t$$

En donde P_r es la potencia recibida, G_t y G_r son la ganancia de las antenas de transmisión y recepción respectivamente, λ es la longitud de onda usada y d la distancia entre el transmisor y receptor.

A partir de esta ecuación se puede despejar la distancia teórica entre ambos nodos.

$$d(L_{dB}) = \sqrt{\frac{G_t \times G_r \times \lambda^2}{4\pi \times P_r} \times P_t}$$

λ se puede obtener a partir de la frecuencia (F), utilizando la velocidad de la luz (C).

$$\lambda = \frac{C}{F}$$

Por otro lado, usando las fórmulas de decibel se puede obtener la potencia (P_1) a partir de la intensidad (L_{dB}) en relación a una potencia de referencia (P_0) que en el caso de las redes inalámbricas es 1mW.

$$P_1 = 10^{\frac{L_{dB}}{10}} \times P_0$$

Finalmente, obtenemos la distancia teórica entre los nodos a partir de la siguiente ecuación.

$$d(L_{dB}) = \sqrt{\frac{G_t \times G_r \times (\frac{C}{F})^2}{4\pi \times 10^{\frac{L_{dB}}{10}} \times P_0} \times P_t}$$

En donde simplificándola y usando los valores antes mencionados se tiene:

$$d(L_{dB}) = \sqrt{\frac{5 \times 5 \times (\frac{299.792.458}{2.442.000.000})^2}{4\pi \times 10^{\frac{L_{dB}}{10}} \times 0,001} \times P_t}$$

$$= \sqrt{\frac{29,98336}{10^{\frac{L_{dB}}{10}}} \times P_t}$$

V. ALGORITMO

A. Tecnologías usadas

Los lenguajes de programación usados son C/C++ [16] [17] debido a su sencillez, bajo consumo de memoria, velocidad de ejecución y herramientas disponibles [18].

Se implementará el algoritmo usando el framework *Malva* [19], el cual es un fork de *Mallba* [20]. Se estudiaron otras alternativas pero finalmente se optó por este debido a recomendaciones hechas en el curso y al hecho de que contábamos con el soporte del foro de discusión del curso.

B. Variante

Luego de estudiar varias alternativas se optó por usar la variante de malva GA, correspondiente a un algoritmo genético genérico con operadores personalizados, con el objetivo de tener total control sobre la aplicación de los mismos sobre los individuos.

C. Representación interna

Para la representación interna del individuo se pueden adoptar dos enfoques distintos, dependiendo de como se interprete la intensidad de las señales leídas por las tarjetas de red inalámbrica del dispositivo.

En un comienzo se implementó el algoritmo utilizando el primero de dichos enfoques, pero luego de un estudio más profundo del modelo de propagación de ondas elegido, se encontró que el segundo enfoque ofrece ventajas sustanciales, por lo que se prosiguió a adaptar el algoritmo de acuerdo a él.

Para entender ambos enfoques es importante primero presentar dos escenarios de ejemplo con el objetivo de explicar de que manera afectan las dos alternativas al problema presentado.



Fig. 1: Escenario ideal



Fig. 2: Escenario real

1) *Escenario*: La imagen 1 presenta un escenario ideal en donde se tiene línea de visión entre el router transmisor y el dispositivo receptor, ubicados a una distancia de 10m uno del otro. La señal percibida en este ejemplo es de de -50dbM.

La imagen 2 presenta un escenario un poco más realista, en donde la línea de visión se ve obstruida por un obstáculo (e.g. una pared) entre ellos. En este caso las ondas se ven perturbadas y el dispositivo solo percibe una señal de -80dbM.



Fig. 3: Escenario percibido

La imagen 3 representa lo que el dispositivo percibiría. Una señal de -80dBm será interpretada por él como una distancia entre el transmisor y receptor de 12m en lugar de 10m, pero esto como no es real. El problema de fondo es la imposibilidad de determinar la existencia de obstáculos entre ambos extremos.

2) *Enfoque 1 - Atenuación Explícita*: Cada individuo es una n-úpla de la forma:

$$I = (x, y, d_1, \dots, d_n)$$

$$x, y \in (-\infty, +\infty)$$

$$d_i \in [0, 1] \forall i \in (1, \dots, n)$$

En donde (x, y) se corresponde con las coordenadas estimadas del individuo en el plano y las constantes d_i representan la atenuación estimada de la señal entre dicho nodo y el router i .

De acuerdo a la representación descrita $(10, -5, 0.15, 0, 0.1)$ representa un individuo en una posición estimada $(10, -5)$ en el plano (con respecto al origen de coordenadas arbitrario usado para inicializar los routers) y que tiene un decaimiento estimado de señal del 15%, 0% y 10% con el router R_1 , R_2 y R_3 respectivamente.

Si la intensidad de la señal que percibe de, por ejemplo, el router R_1 es de 82%, para el cálculo del fitness se asumiría que la misma debió ser 15% mayor, por lo cual se usará una intensidad teórica de $82 \times 1.15 = 94\%$ en su lugar.

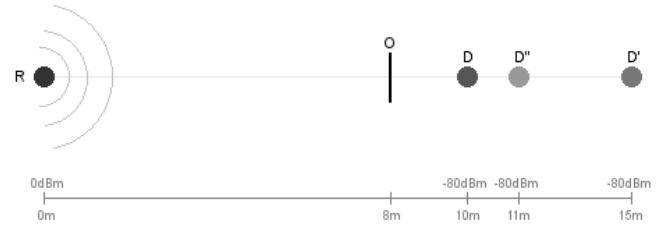


Fig. 4: Escenario mejorado

La imagen 4 muestra el efecto que tiene sobre la estimación de la posición el hecho de considerar la atenuación hacia cada router.

Este enfoque, si bien en un primero momento parecía adecuado, tiene un defecto importante. La representación interna del individuo contiene un valor de atenuación hacia cada uno de los routers, por lo que dado cualquier valor para (x, y) , el algoritmo será capaz de modificar dichos valores de manera que el fitness sea bueno. En otras palabras, sin importar la estimación de la posición, existirá un vector de atenuación que haga que el individuo sea aceptable, potencialmente mejor que otro cuya posición estimada es más acertada. Asimismo, la representación interna dependería directamente de la cantidad de routers usados ya que para cada uno de ellos se deberá estimar un decaimiento, ocasionando que rápidamente el espacio de búsqueda crezca al punto de que no se explore lo suficiente en tiempos razonables.

3) *Enfoque 2 - Atenuación Implícita*: Cada individuo es un par de la forma:

$$I = (x, y)$$

$$x, y \in (-\infty, +\infty)$$

En donde (x, y) se corresponde con las coordenadas estimadas del individuo en el plano.

Este segundo enfoque extiende al anterior haciendo un uso más exhaustivo de la información obtenida del modelo de propagación de ondas. La idea a grandes rasgos consiste en inferir cuál es el decaimiento debido a los obstáculos, de manera que no sea necesario que el algoritmo los estime.

Para esto es importante notar que dado cualquier punto en el plano, existen dos valores de intensidad de señal. Por un lado se tiene la *señal teórica*, que corresponde a la señal que debería estar llegando en un escenario ideal y queda determinada únicamente a partir de la distancia entre el transmisor y el receptor. Por otro lado se tiene la *señal recibida*, que corresponde a la *señal teórica* menos un *decaimiento* debido a obstáculos e interferencias.

De esta manera, la diferencia de ambas magnitudes es una buena aproximación del decaimiento usado en el enfoque anterior.

$$d_i = s(d) - s$$

Este enfoque no requiere una representación interna dependiente de la cantidad de routers y solo se limita a ajustar la posición del dispositivo, logrando un espacio de búsqueda mucho más reducido, aunque infinito.

La idea de aquí en más es adoptar este enfoque y minimizar el vector de decaimientos inferido. En otras palabras, el algoritmo tenderá a acercar los dispositivos a la distancia teórica que corresponda con la intensidad de señal recibida.

Es importante recordar que esto se estará haciendo para más de un router al mismo tiempo, por lo que si bien el vector del óptimo global será mínimo, en principio no será nulo.

D. Función de fitness

La función de fitness a minimizar es el vector de decaimiento calculado a partir de la posición estimada (usando el modelo de propagación de ondas *free space propagation model* y la intensidad de señal recibida.

$$f(x, y) = \sum_{i=1}^n |\sqrt{(R_{i_x} - x)^2 + (R_{i_y} - y)^2} - d(R_{i_s})|$$

En donde (R_{i_x}, R_{i_y}) y R_{i_s} son la posición y la intensidad de señal recibida del router i respectivamente, mientras que (x, y) es la posición estimada del dispositivo que se está triangulando.

La función de fitness expresada de esta forma es una aproximación de la suma de los errores reales a partir de la posición estimada. En efecto, lo que se está sumado son las diferencias entre la distancia estimada a un router y la distancia teórica obtenida mediante el modelo de propagación.

E. Inicialización

Como se ha dicho antes, se asume que los routers comparten características similares, en particular, que el alcance máximo para todos ellos es de A metros.

La población será inicializada eligiendo valores aleatorios para (x, y) de manera que todos los individuos comiencen en la intersección de los círculos de centro (R_{i_x}, R_{i_y}) y radio A , en donde (R_{i_x}, R_{i_y}) es la posición del router R_i .

De esta manera, durante la inicialización de la población, los valores estarán limitados por:

$$x_{max} = \text{rand}(\max_{i=1} R_{i_x} - A, \min_{i=1} R_{i_x} + A)$$

$$y_{max} = \text{rand}(\max_{i=1} R_{i_y} - A, \min_{i=1} R_{i_y} + A)$$

En donde $\text{rand}(\min, \max)$ es una función que devuelve un número en punto flotante aleatorio entre \min y \max .

F. Operadores

Dado que el espacio de búsqueda es inmenso, resulta esencial mantener la diversidad hasta haber muestreado una amplia área del plano, ya que en principio se puede asumir la existencia de muchos óptimos locales. Dicho esto, los operadores que serán utilizados se describen a continuación.

1) *Selección*: Se usará selección estocástica universal con punteros equiespaciados.

2) *Cruzamiento*: El operador de cruzamiento promediará los valores de ambos padres.

$$x_3 = \frac{x_1 + x_2}{2}, y_3 = \frac{y_1 + y_2}{2}$$

3) *Mutación*: El operador de mutación alterará de acuerdo a un valor aleatorio elegido en un determinado rango.

$$x = x + k_1, y = y + k_2, k_1, k_2 \in (n, m)$$

4) *Búsqueda Local*: Dadas las magnitudes de las distancias con las que se trabaja, un error de 0cm, 25cm o 50cm son igual de aceptables. Por lo tanto no existe la necesidad de optimizar el individuo para alcanzar la mejor estimación posible en su entorno. Se considerará que un individuo con errores de estos ordenes es suficientemente bueno para el problema de modo que no será necesario intentar disminuir aún más el error.

G. Criterio de parada

Las distancias con las que se trabajan son del orden de unos pocos metros a algunas decenas de metros. A partir de esto parece razonable parar el algoritmo si se consigue un individuo cuya suma de errores (es decir, el valor de la función de fitness) es menor a 50cm.

Al mismo tiempo será necesario un límite en el número de generaciones con el objetivo de evitar que el algoritmo converja a un óptimo local en donde no puede lograr tal precisión, ocasionando que no termine de ejecutarse. Para evitar esto empíricamente se encontró que un límite de 500 generaciones es suficiente, ya que en la mayoría de las ejecuciones la generación del mejor individuo estaba por debajo de este límite.

H. Mecanismo de reparación

Cabe señalar que los individuos son válidos desde el momento de su inicialización y los operadores aplicados no los corrompen, por lo que no hace falta un mecanismo de reparación o de descarte de individuos inválidos.

VI. ANÁLISIS EXPERIMENTAL

A. Entorno de prueba

Dado que en la práctica la implementación realizada no parece necesitar de grandes tiempos de ejecución ni grandes cantidades de memoria o procesamiento que requieran correrlo en un cluster, se usará una computadora de características moderadas para realizar las pruebas.

1) *Marca y modelo:* Dell Inspiron N5010

2) *Procesador:* Intel Core i5 2.67GHz (doble núcleo, cuatro threads).

64KB de cache L1 512KB de cache L2 3MB de cache L3 4GB de memoria DDR3 1333MHz

3) *Sistema Operativo:* Se usó *GNU/Linux* [21] [22], en su variante *Debian 7.0 (Wheezy)* [23] para 64 bits (amd64).

Se prestó especial cuidado a los procesos corriendo, deteniendo todos los procesos y servicios innecesarios al momento de realizar las pruebas.

B. Instancias de prueba

Para el ajuste paramétrico se usará una instancia de prueba reducida, conformada por 3 routers ubicados aleatoriamente en el plano.

Para realizar la evaluación del algoritmo se usarán 25 ejecuciones divididas en 5 conjuntos de 3, 4, 5, 6 y 7 routers (cada conjunto contendrá 5 escenarios inicializados independientemente).

Además de estas 25 ejecuciones se realizará una prueba real con 3 routers ubicados entre 3 y 10 metros de la computadora de prueba, con el objetivo de validar la implementación en un escenario real.

C. Ajuste paramétrico

La tabla I muestra las combinaciones de parámetros estudiados para hacer el ajuste paramétrico. Para cada combinación se realizaron 3 ejecuciones, registrando el valor promedio de tiempo de ejecución (en segundos) y fitness del mejor individuo.

TABLE I: Ajuste paramétrico

| Población | Cruzamiento | Mutación | Tiempo | Fitness |
|-----------|-------------|----------|--------|---------|
| 50 | 0.85 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.90 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.95 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| 75 | 0.85 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.90 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.95 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| 100 | 0.85 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.90 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |
| | 0.95 | 0.01 | 0.0 | 000 |
| | | 0.05 | 0.0 | 000 |
| | | 0.10 | 0.0 | 000 |

Análisis: ...

D. Evaluación

Para evaluar el algoritmo se harán 30 ejecuciones independientes para el caso de 3, 4, 5, ..., 10 nodos. Adicionalmente se harán 3 ejecuciones en un entorno real, usando 3, 4 y 5 routers.

Para todos los casos se reportará:

- Tiempo de ejecución
- Desviación estándar de los tiempos de ejecución
- Fitness del mejor individuo
- Generación del mejor individuo
- Valor promedio del fitness
- Desviación estándar de los mejores fitness

Para los casos reales se reportará además:

- El error entre la posición del mejor individuo y la posición real que debió encontrar
- La diferencia entre dicha posición y la posición devuelta por servicios existentes

Análisis: ...

E. Análisis de normalidad

...

TABLE II: Resultados

| 3 Routers | | | | | |
|------------|---------|----------|------------|--------|------------|
| Generación | Fitness | Promedio | Desviación | Tiempo | Desviación |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |

| 4 Routers | | | | | |
|------------|---------|----------|------------|--------|------------|
| Generación | Fitness | Promedio | Desviación | Tiempo | Desviación |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |

| 5 Routers | | | | | |
|------------|---------|----------|------------|--------|------------|
| Generación | Fitness | Promedio | Desviación | Tiempo | Desviación |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |

| 6 Routers | | | | | |
|------------|---------|----------|------------|--------|------------|
| Generación | Fitness | Promedio | Desviación | Tiempo | Desviación |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |

| 7 Routers | | | | | |
|------------|---------|----------|------------|--------|------------|
| Generación | Fitness | Promedio | Desviación | Tiempo | Desviación |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |
| 000 | 0.00 | 0.00 | 0.00 | 000 | 000 |

VII. CONCLUSIÓN

The conclusion goes here.

REFERENCES

- [1] Skyhook's mapped wifi access points. <http://www.skyhookwireless.com/location-technology/coverage.php>.
- [2] Wifi positioning system. http://en.wikipedia.org/wiki/Wi-Fi_positioning_system.
- [3] Google maps geolocation api. <https://developers.google.com/maps/documentation/business/geolocation>.
- [4] Skyhook location sdk. <http://www.skyhookwireless.com/location-technology>.
- [5] M. A. Ansari, S. Farooqui, and A. Yasar. Low cost solution for location determination of mobile nodes in a wireless local area network. pages 1–7, 2005.
- [6] T. Garcia-Valverde, A. Garcia-Sola, H. Hagra, J. Dooley, V. Callaghan, and J. Botia. A fuzzy logic based system for indoor localisation using wifi in ambient intelligent environments. *Fuzzy Systems, IEEE Transactions on*, PP(99):1–1, 2012.
- [7] Sheng-Cheng Yeh, Wu-Hsiao Hsu, and Yih-Shyh Chiou. Adaptive-weighting schemes for location-based services over heterogeneous wireless networks. pages 1–4, 2010.
- [8] P. Bahl and V.N. Padmanabhan. Radar: an in-building rf-based user location and tracking system. 2:775–784 vol.2, 2000.
- [9] M. A. Ansari, S. Farooqui, and A. Yasar. Low cost solution for location determination of mobile nodes in a wireless local area network. pages 1–7, 2005.
- [10] R. Ezzine, A. Al-Fuqaha, R. Braham, and A. Belghith. A new generic model for signal propagation in wi-fi and wimax environments. pages 1–5, 2008.
- [11] S.Y. Seidel and T.S. Rappaport. 914 mhz path loss prediction models for indoor wireless communications in multifloored buildings. *Antennas and Propagation, IEEE Transactions on*, 40(2):207–217, 1992.
- [12] D. P. Agrawal and Q-A. Zeng. *Introduction to Wireless and Mobile Systems*. Thomson, 2006.
- [13] T.K. Sarkar, Zhong Ji, Kyungjung Kim, A. Medouri, and M. Salazar-Palma. A survey of various propagation models for mobile communication. *Antennas and Propagation Magazine, IEEE*, 45(3):51–82, 2003.
- [14] Theodore S. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, 2002.
- [15] Harald T. Friis. Introduction to radio and radio antennas. *Spectrum, IEEE*, 8(4):55–61, 1971.
- [16] The c resources network. <http://www.cprogramming.com/>.
- [17] The c++ resources network. <http://www.cplusplus.com/>.
- [18] D.A. Cargill and M. Radaideh. A practitioner report on the evaluation of the performance of the c, c++ and java compilers on the os/390 platform. pages 40–45, 2000.
- [19] Malva. <https://code.google.com/p/malva/>.
- [20] Gabriel Luque. Mallba library v2.0. <http://neo.lcc.uma.es/mallba/easy-mallba/>, 2006.
- [21] The gnu operating system. <http://www.gnu.org/>.
- [22] The linux kernel archives. <https://www.kernel.org/>.
- [23] Debian wheezy release information. <http://www.debian.org/releases/stable/>.