VNIVERSIDAD NACIONAL AVPNOMA DE MEXICO

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

PROGRAMA DE MAESTRIA Y DOCTORADO EN INGENIERIA

FACULTAD DE INGENIERÍA

TÉCNICAS DE PRIVACIDAD GEOGRÁFICA EN REDES MÓVILES

T E S I S

QUE PARA OPTAR POR EL GRADO DE:

MAESTRO EN INGENIERIA

INGENIERÍA ELÉCTRICA - TELECOMUNICACIONES

P R E S E N T A :

ING. OSCAR ARANA HERNÁNDEZ

TUTOR:
DR. JAVIER GÓMEZ CASTELLANOS

2010

JURADO ASIGNADO:

Presidente:	DR. GARCÍA UGALDE FRANCISCO
Secretario:	DR. RANGEL LICEA VÍCTOR
Vocal:	DR. GÓMEZ CASTELLANOS JAVIER
1er. Suplente:	DR. MOCTEZUMA FLORES MIGUEL
2do. Suplente:	DR. PASCOE CHALKE MIICHAEL

Lugar o lugares donde se realizó la tesis:

CIUDAD UNIVERSITARIA, MÉXICO, D.F.

TUTOR DE TESIS:

DR. JAVIER GÓMEZ CASTELLANOS

_____

FIRMA

Dedicatoria

A mi familia, a Raul mi padre y Gisela mi madre sin quienes obviamente, no estaría aquí.

A mis hermanos, Raul y Erin, ni modo les tocó ser mis hermanos.

A mi mujer, Stephany:

"... el estado de enamoramiento, lejos de cerrar el camino de la pasión, más bien parece abrirlo; lejos de ser vía de ceguera y tinieblas, es vía de lucidez y conocimiento..."

Eugenio Trías, Tratado de la pasión.

Gracias mi amor por estar siempre conmigo.

Agradecimientos

A la Universidad Nacional Autónoma de México mi Alma Máter.

Al Posgrado de Ingeniería y a la Coordinación de Estudios de Posgrado, por el apoyo económico que me permitió realizar los estudios de maestría.

Al Dr. Javier Gómez C. por su trabajo y apoyo incondicional durante todo este proceso y se ven reflejados en esta tesis.

A los proyectos PAPIIT 106609 y CONACyT 105117.

A mis sinodales por su cooperación y sus correcciones.

Finalmente a todos mis amigos sin cuyos consejos y apoyo esta tesis no hubiera sido la misma:

Gonzalo, Isabel, Cintya, Carlos Genaro y Axel.

Resumen

En la actualidad el abaratamiento de tecnologías tales como telefonía celular, sistemas de posicionamiento global y redes de acceso inalámbrico, han permitido el rápido desarrollo e introducción al mercado de servicios basados en la localización. Dichos servicios, hacen uso del equipo de comunicaciones así como de sistemas de localización para brindarles a los usuarios sugerencias acerca de algunos sitios de su interés.

El crecimiento de dichos servicios podría involucrar de una serie de riesgos para los usuarios finales. Su implantación sin ninguna medida de control, en cuanto a la información que proporcionan los usuarios, podría ocasionar una fuga importante de información que podría poner en riesgo la integridad de sus usuarios.

Es por esto que se han definido algunas estrategias para que los usuarios tengan mayor control sobre la información de su localización. En esta tesis exploramos algunos trabajos acerca de la definición e implementación de algunas de estas técnicas, así como también presentamos dos técnicas de ofuscación que ayuden a mitigar los efectos de la estimación indeseada en la trayectoria de los usuarios finales.

Estas dos técnicas se desarrollan bajo el concepto de que los usuarios sean los que decidan cuándo y cómo implementarlas ya que, como veremos más adelante, existen diferentes requerimientos de privacidad.

Así mismo, se llevarán a cabo simulaciones de los efectos de las dos técnicas de ofuscación, aquí presentadas, dentro de un simulador que implemente un algoritmo que estime las trayectorias de los nodos móviles. Esto nos permitirá comprobar los efectos de las técnicas.

Abstract

Currently, the low price of technologies such as cell phones, global position systems and wireless networks has allowed the fast development of location based services and its introduction to the market. These services make use of communication equipment as global position system to give users suggestions about some sites of their interest.

The increase of such services could involve a series of risks for final users. The implementation of these services without any measures to control the information that users provide could produce an important information leakage, which would pose a risk to the integrity of users.

Therefore, some strategies have been defined, so that users can have better control over their location data. In this research work, we explore other papers related to the definition and implementation of some of these techniques. We also present two techniques of bewilderment that help reduce the effects of unwanted estimates in the trajectory of final users.

These two techniques are developed with the aim that the users decide when and how to implement them, because as we will see later, there are several different privacy requirements.

Besides, we will carry out simulations of the effects of the two techniques of bewilderment in a simulator which implements an algorithm to estimate the trajectories of the moving nodes. This will allow us to test the effects of the techniques.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Definición del Problema

El inicio de la convergencia en la infraestructura de las telecomunicaciones, así como el surgimiento de dispositivos de cómputo móvil con mayores capacidades de conexión, ha permitido una revolución en la manera en la que interactuamos con el mundo.

En la actualidad vivimos una época de hiperconectividad, según Nortel Networks, la hiperconectividad se define como *"... el estado en el cual el número de dispositivos, nodos y aplicaciones que están vinculadas con la Red exceden ampliamente el número de personas conectadas. Este fenómeno ya está ocurriendo y se espera que en 2010 cada persona tenga alrededor de diez dispositivos conectados ..."*[1].

Si bien dicha proyección no ha sido alcanzada, o por lo menos no en nuestro país, tampoco nos encontramos muy alejados de ella; todos los días una gran cantidad de nuevos dispositivos móviles salen al mercado, la diversidad de dichos dispositivos también ha aumentado, actualmente podemos encontrar: teléfonos celulares, Asistente Digital Personal (Personal Digital Assistant) (PDA), laptops, iPods, cámaras fotográficas y de video, sensores, identificación por radiofrecuencia (Radio Frequency IDentification) (RFID), automóviles, equipos médicos, aparatos domésticos, maquinaria industrial y hasta sistemas de irrigación de campos, los cuales cuentan con capacidades de conexión múltiples. Cada uno equipado con uno o varios radios de comunicación inalámbrica tales como radios 3G con Acceso múltiple por división de código de banda ancha (Wideband Code Division Multiple Access) (WCDMA) en caso de teléfonos celulares de tercera generación, radios Bluetooth y Tecnología de comunicación inalámbrica de datos, empleada en redes de área local (Wi-Fi) (Red de área local inalámbrica (Wireless Local Area Network) (WLAN) identificada por el estándar del Instituto de Ingenieros Electricistas y Electrónicos (Institute of Electrical and Electronics Engineers) (IEEE) 802.11) en los equipos celulares, pero también en Laptops, iPods y PDAs, inclusive podemos encontrar dispositivos con sistema de posicionamiento global (Global Position System) (GPS) en automóviles, PDAs y Celulares.

Así mismo, se observa un crecimiento en el número de horas que una persona pasa conectado a la Internet. Antes sólo se conectaban en su oficina o en las escuelas, ahora también usan parte de su tiempo en casa o algunas veces fuera de ella para estar conectado a la Internet. Como podemos ver en la Figura 1.1, ha ido en aumento tanto el número de usuarios de Internet desde sus hogares como el de usuarios con acceso fuera de sus hogares. Más aún, ya no es difícil obtener acceso Wi-Fi en muchos lugares públicos como aeropuertos, cafés,

tiendas departamentales y restaurantes. Con lo que la WLAN se ha ido convirtiendo en la tecnología predilecta de última milla, es decir la que nos proporciona el acceso hacia Internet, lo que nos ha llevado al inicio del desarrollo de las redes metropolitanas.

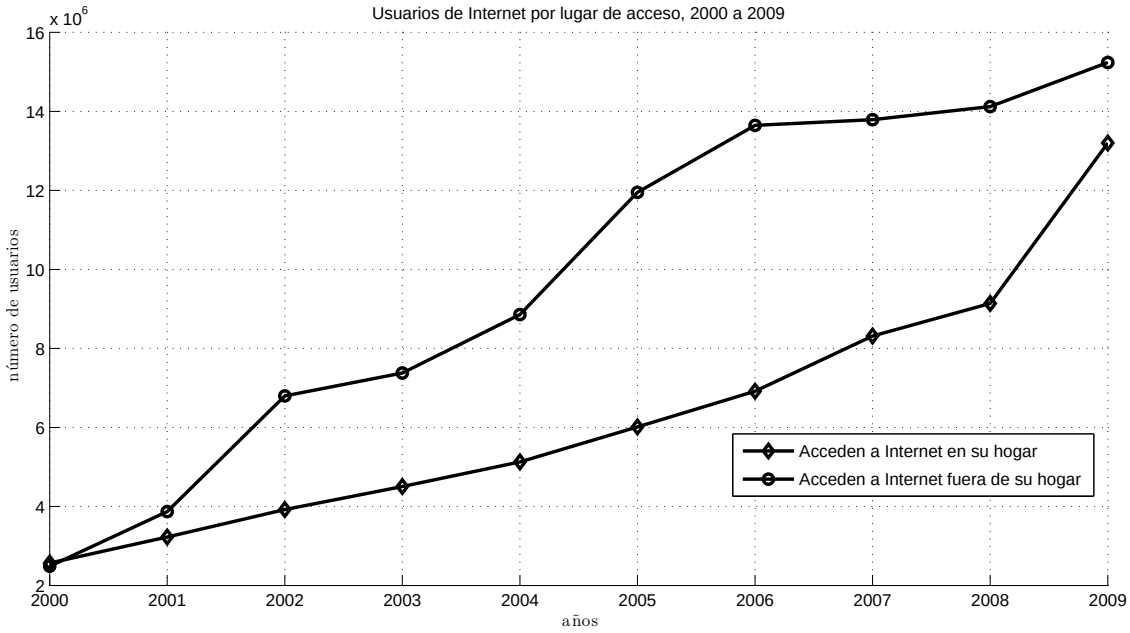


Figura 1.1: Usuarios de Internet por lugar de acceso, 2000 a 2009, [2].

Como su nombre lo indica, las redes metropolitanas nos brindarán acceso a la Internet dentro de las ciudades. Esto se traduce en la capacidad de poder estar conectados a la Internet mientras nos desplazamos y realizamos nuestras actividades rutinarias tales como ir a la escuela, ir al trabajo, ir de compras, o simplemente salir a divertirse. Lo cual nos da un panorama del impacto que tendría la implementación de técnicas de localización basadas en tecnologías de acceso inalámbricas, por ejemplo WLAN y celular (3G).

El problema principal de las técnicas de posicionamiento geográfico radica en que con una cierta cantidad de información obtenida, pueden inferirse algunos patrones de comportamiento o simplemente vulnerar los hábitos del sujeto observado, incluso se pueden brindar datos tales como dónde vive, dónde trabaja, cuánto tiempo se encuentra en su casa, o cuánto en el trabajo, qué días frecuenta que lugares, etcétera.

1.1. Problemas en la privacidad de la localización

Este futuro de hiperconectivdad y redes metropolitanas, parece ser completamente alentador, al poder vivir en una sociedad de la información, con capacidades masivas de comunicación en tiempo real. Esto sólo parece prometer un mundo nuevo de aplicaciones apoyadas en dichas capacidades. Sin embargo, hay que considerar que vivir en dicho entorno no sólo nos trae los beneficios del acceso a la información o de la comunicación, sino también trae consigo riesgos originados por la cantidad de información que nosotros le proporcionamos a las aplicaciones, a las diferentes redes a través de las cuales nos conectamos. Pero sobre todo el principal riesgo es: quién o qué mantiene nuestra información en buenas manos.

Al mismo tiempo que nuestros dispositivos de conexión inalámbrica mejoran y abaratan sus precios, se desarrollan aplicaciones o mejor dicho servicios que hacen uso de las capacidades mejoradas. Por ejemplo, tener un celular con GPS y salida a Internet nos abre un mundo de posibilidades. Podemos solicitar al servicio que nos de la ubicación de los restaurantes de comida china a no más de cien metros de nuestra posición actual. Estos servicios reciben el nombre de Location-Based Services (Servicios Basados en la Localización). Este tipo de servicios explotan el conocimiento de la posición geográfica para ofrecer servicios e información (Internet y/o bases de datos) relativos a hospitales, delegaciones, bloqueos viales, o simplemente lugares de entretenimiento cerca de su posición actual.

Las ventajas de este tipo de servicios, tanto comerciales como sociales, son muchísimas. Esto porque simplifican la búsqueda de información para el usuario final. En este momento, las tres compañías más grandes de telefonía celular en México ya comienzan a ofrecer estos servicios. En un primer acercamiento, la localización está enfocada a empresas para el control de flotillas. En el ámbito comercial, estos servicios estarán orientados a conocer la ubicación aproximada de amigos y familiares que así lo deseen, así como las ofertas en centros comerciales próximos a la posición del usuario y también para recibir notificaciones de bloqueos viales cercanos.

Debemos recordar que si bien estos servicios pueden mejoran nuestra calidad de vida, vienen acompañados de nuevos retos y amenazas, pero sobre todo no debemos de permitir que se intercambien los beneficios por nuestra privacidad y sobre todo nuestra seguridad.

Esto nos lleva a pensar en qué tanta protección a la privacidad geográfica es necesaria, sabemos que la privacidad perfecta no puede lograrse mientras tengamos la necesidad de comunicarnos a través de dichas redes. El nivel requerido de esta protección no debe ser un asunto de tecnología, diferentes personas tendrán diferentes necesidades de privacidad. Sin perder de vista que "Nunca la tecnología debe forzar a la sociedad a aceptar menos privacidad"[4].

La preocupación por la privacidad no es nueva, siempre ha existido cierta capacidad de rastrear o seguir a las personas. Sin embargo, esto resultaba complicado, llevaba mucho tiempo poder reunir la información necesaria para formar un perfil significativo acerca de

una persona. Dichas desventajas parecen desaparecer si consideramos el alcance de nuestros dispositivos personales. La tecnología si bien permite una mayor comunicación, también puede permitir la existencia de dispositivos dedicados que vigilen por más tiempo a las personas, posibilitando así reunir la cantidad de información suficiente para deducir los hábitos de las personas, sus historiales, en menos tiempo.

Supongamos entonces que una persona tiene un teléfono celular, muchas de sus actividades cotidianas están ligadas al uso de este dispositivo, el cual permanece siempre encendido y comunicándose con las radio bases a su alcance. En una ciudad, como la Ciudad de México, las compañías telefónicas se encargan de mantener la cobertura y que al mismo tiempo, dicha cobertura, sea lo más extensa posible. Por lo que, mientras esta persona permanezca dentro de la ciudad, estará siempre dentro de la cobertura de su compañía telefónica. Un objetivo que nosotros consideramos deseable, ya que pagamos por el servicio y queremos estar siempre comunicados.

Qué pasa ahora, si un tercero tiene acceso al registro de la compañía telefónica, donde se indican las coordenadas geográficas de las radio bases en las que dicha persona se encuentra registrada actualmente. También se encuentran en dicho registro las lecturas de parámetros como la potencia y el ID de usuario, Más aún, tiene acceso a los archivos históricos en los que dicha persona, aparece junto con las coordenadas de las radio bases por las que pasó a lo largo de algunos días y cuanto tiempo estuvo esa persona registrado en ciertas radio bases. Sería entonces posible relacionar la información para saber por ejemplo que la mayoría de las noches su teléfono celular queda registrado en determinada radio base, dicha persona entonces ha revelado la posible ubicación de su casa. Así mismo, la información indica que permanece cerca de ciertas radio bases, las cuales se encuentran próximas a una zona escolar entre las siete y ocho de la mañana, es posible que usted tenga hijos y los lleve a dichas escuelas, también se puede saber donde trabaja dado que permanece registrado en alguna radio base durante el periodo de horas laborales en un día.

Supongamos ahora un caso diferente, supongamos que una persona tiene un perfil médico el cual requiere que reciba atención médica inmediata bajo ciertas condiciones o en ciertos eventos, dicha persona requiere entonces de un monitoreo geográfico intensivo, dado el peligro de no poder comunicar su ubicación a tiempo. Actualmente, en E.E.U.U. se trabaja en un servicio denominado E911, este servicio consiste en poder rastrear geográficamente a la persona que realize una llamada a los servicios de emergencia, esto mediante la información proporcionada por la red móvil de comunicación a través de la cual se realizó la llamada y así poder asistirlo. Como hemos visto diferentes casos de personas con requerimientos distintos de privacidad, entonces ¿quién debe ser el encargado de proporcionar dichos niveles de privacidad y asegurar que se cumplan?.

En [5]Duckham y Kulik hacen una revisión de la privacidad y proponen cuatro estrategias para procurar la privacidad en la localización.

- Estrategias regulatorias: Por lo regular, el gobierno es la entidad que debe establecer normas para el uso adecuado de la información personal.

- Políticas de privacidad: Estas políticas se deben establecer entre el usuario de dichos servicios y quien sea que está proveyendo los servicios y usando la información de dicho usuario.

- Anonimato: El usuario hace uso de datos falsos de identidad y crea ambigüedades de información al agruparse con otros usuarios.

- Ofuscación: El usuario reducirá la calidad de la información que será usada para obtener su posición geográfica.

Podemos notar que las dos primeras requieren de un organismo en el cual se deposite la confianza de los usuarios, mientras que las dos últimas requieren que el usuario se haga cargo por si mismo de mantener el nivel de privacidad requerido. En la presente tesis nos enfocaremos sólo en la tercera y cuarta estrategia.

Como parte de este capítulo definiremos los objetivos correspondientes a la presente tesis.

1.2. Objetivos generales

- Implementar un ambiente de simulación que permita cuantificar los efectos de las técnicas de ofuscación aplicadas sobre el algoritmo de localización.

- Definir e implementar en el ambiente de simulación un algoritmo de localización que permita determinar la posición geográfica de los nodos móviles dentro del área de cobertura de las redes de acceso inalámbrico.

- Definir e implementar en el ambiente de simulación dos técnicas de ofuscación en la localización.

    - Control de potencia.
    - Creación de nodos virtuales.

1.3. Metodología

La investigación se llevó a cabo mediante el uso del software matemático MatLab para construir un ambiente de simulación, el cual nos permitiera implementar tanto las técnicas de localización, como las de anti-localización y cuantificar los efectos de estas últimas sobre las primeras.

1.4. Contribuciones

En la presente tesis, proponemos una técnica para poder estimar la localización de nodos inalámbricos dentro de la zona de cobertura de su respectiva red de comunicaciones. Esta técnica convierte el problema de localización en una serie de casos geométricos, donde podemos aplicar principios trigonométricos básicos para poder determinar la posición. Así mismo proponemos dos nuevas técnicas de ofuscación que permitan proteger la información de la localización de nodos inalámbricos. Las dos nuevas técnicas de ofuscación propuestas, estan basadas en parámetros comunes a cualquier sistema de comunicaciones inalámbrico, por lo que podrían ser implementados en cualquier sistema de comunicaciones.

1.5. Estructura de la tesis

La tesis se organiza como sigue: En el segundo capítulo hacemos una revisión de los artículos internacionales más relevantes con respecto a las de técnicas de localización y antilocalización. En el tercer capítulo desarrollaremos nuestro algoritmo de estimación de la trayectoria. En el cuarto capítulo definiremos las dos técnicas de ofuscación presentadas en esta tesis. En el quinto capítulo llevaremos a cabo la implementación de las dos técnicas de ofuscación presentadas en el software MatLab. En el sexto capítulo presentaremos las pruebas y resultados que realizamos. En el séptimo y último capítulo expondremos las conclusiones de esta tesis.

Capítulo 2

Antecedentes

2.1. Estado del arte en las técnicas de localización

Uno de los primeros sistemas especialmente diseñado para el Tracking (estimación de la trayectoria) geográfico fue el GPS. Este sistema hace uso de satélites para ayudar a los dispositivos a determinar su posición. Dicho sistema requiere al menos de la señal de tres satélites para poder determinar su posición mendiante la triangulación tomando en cuenta la posición de dichos satélites. Comercialmente puede alcanzar una resolución de más o menos cuatro metros. Actualmente es ampliamente usado e implementado en dispositivos móviles como teléfonos celulares y PDAs, así como en automóviles.

Sin embargo, dicho sistema sólo funciona de manera adecuada dentro del alcance de los satélites, es decir, teniendo línea de vista. Algunos investigadores han explotado ese aparente defecto de la tecnología para esquemas nuevos de localización, por ejemplo Marmasse y Schmandt's en [6] definen el sistema "comMotion", el cual marca como significativo un lugar en el que se ha perdido la señal de GPS tres o más veces dentro de un radio determinado. Más adelante Marmasse continuó trabajando con este esquema, comenzó a combinar los tiempos en los que el usuario desaparece del alcance de los satélites, debido al bloqueo de los edificios o periodos de poco alcance de los mismos, para establecer lugares potencialmente significativos para los usuarios.

Hariharan y Toyama en [7] definen un sistema basado en segmentación en tiempo y sensible a la locación, para representar jerárquicamente los lugares donde el usuario permanece y a los que se dirige. Hightower en [8] muestra el algoritmo BeaconPrint, el cual encuentra conjuntos repetibles de radio bases GSM y Wi-Fi, a las que los usuarios de dichas tecnologías suele aproximarse o detenerse. Este enfoque resulta interesante ya que no usa coordenadas espaciales como indicadores de la posición, sino que se encarga sólo de escuchar transmisores de radio.

Beresford y Stanjano en [3] muestran como, tomando medidas de localización obtenidas de su sistema de posicionamiento ultrasónico de baja potencia (Active Bat) en ambientes cerrados, fueron capaces de inferir los hábitos de las personas que ahí laboraban, enfocándose en encontrar donde las personas pasaban más tiempo. Por último combinaron esa información con los nombres de los trabajadores tomando en cuenta las posiciones de los escritorios dentro

del edificio.

Hoh en [9], hace uso de una base de datos de posiciones GPS con lecturas de toda una semana proporcionadas por 239 choferes de taxis en la ciudad de Detroit. Examinó un subconjunto de 65 choferes y fue capaz de encontrar la ubicación de sus casas en un 85 % de los casos. Esto a pesar de que los autores no contaban con información previa acerca de los choferes. Un ataque similar realizado por Krumm en [10], utilizó bases de datos de dos semanas de duración, con lecturas de GPS obtenidas de 172 conductores. Este estudio también tuvo como objetivo determinar las coordenadas geográficas de las casas de los conductores, al comparar los resultados obtenidos con las ubicaciones reales se encontró un error aproximado de 61 metros. Para refinar los resultados se realizó una búsqueda posterior en los directorios de la sección blanca con dichas ubicaciones aproximadas y se encontró la posición exacta de las casas en un 13 % de los casos mientras que en el 5 % de los casos, se pudo determinar el nombre de los conductores.

Posteriormente Gruteser y Hoh en [11], trabajaron con datos GPS que fueron completamente anónimos, esto en el sentido de que ningún pseudónimo fue proporcionado con los registros de tiempo, coordenadas de latitud y longitud. Ellos implementaron una técnica estándar de Tracking Multi-target, o bien rastreo multi objetivo [12], para separar adecuadamente los datos GPS de tres personas diferentes. Esto demostró que incluso mezclando las posiciones entre los usuarios es posible re ensamblar de manera coherente la información y dividirla en cada una de las tres trayectorias.

Más allá del uso de dispositivos GPS, muchos investigadores han intentado demostrar que sin estos dispositivos puede llevarse a cabo el rastreo de las personas. Estos ejemplos los mostraré a continuación.

Wilson y Atkenson en [13] ubicaron sensores de proximidad en una casa, entre los cuales se encontraban sensores de movimiento, de presión, switches de contacto etcétera. El disparo de cada uno de los dos estados de dichos dispositivos quedaba registrado, y dada la naturaleza de los sensores, no brindaban información de quien era la persona que los habia activado. Con la información proporcionada por los miles de eventos registrados por los sensores, se desarrolló un algoritmo probabilístico para interpretar la información, dicho algoritmo acertó en un 85 % de las asociaciones con cada uno de los habitantes. Este algoritmo estima, con una cierta probabilidad, cual de los tres ocupantes se encuentra en alguna parte de la casa en un momento dado, con base en la lista de los miles de registros de los disparos de los sensores. Las implicaciones de este trabajo son fundamentales ya que dejan entrever la vulnerabilidad de los hábitos de las personas. Con una pequeña cantidad de información obtenida se pudieron estimar sus ubicaciones, pero sobre todo las ubicaciones vinculadas a un tiempo, con lo que podría llevarse a cabo la construcción de historiales.

En esta sección pudimos revisar, sino todos los ejemplos de algoritmos de localización desarrollados, sí mencionamos los más significativos para este trabajo.

2.2. Estado del arte en las técnicas de anti-localización

Como mencionamos previamente, Duckham y Kulik definieron cuatro estrategias para conseguir la privacidad, en resumen, 2 asúmen que se debe confiar en terceros (Estrategias regulatorias y Políticas de privacidad) para conseguir la privacidad deseada, mientras que las otras dos estrategias (anonimato y ofuscación) asumen al usuario como el que se procure la privacidad así mismo, sin hacer uso de organismos depositarios de la responsabilidad.

En esta tesis nos concentraremos en este segundo enfoque, que el usuario se haga cargo por sí mismo de obtener su nivel de privacidad adecuado. Esto nos lleva a redefinir la privacidad en la ubicación (Location Privacy). Beresford y Stanjano en [3] definen la privacidad en la ubicación como *la habilidad de prevenir que terceros sepan nuestra posición actual o pasada.*

Esta definición nos transmite la idea de que una persona la cual está siendo rastreada debería tener el control de quien quiere que se entere de esta información o bien, de alguna manera, mantener el control en la precisión con la cual será obtenida su ubicación geográfica. Por otra parte también indica la importancia no sólo de la posición actual, sino también de las posiciones anteriores. Mientras que la obtención de la posición en tiempo real puede permitir a un atacante encontrar al objetivo, la información pasada puede permitirle al mismo atacante averiguar mucha más información sobre el objetivo, información como: quién es, dónde vive, y a qué se dedica o en qué trabaja.

En [5] Duckham y Kulik redefinen el concepto de privacidad en la localización como: *un tipo especial de aislamiento de la información mediante el cual le corresponde sólo a los individuos determinar cuándo, cómo y para qué se brinda información de ellos a las demás personas.*

Esta definición tiene implícita la importancia de revelar la información de la ubicación de diferentes maneras, entre las cuales podemos encontrar diferentes esquemas, tales como: el uso de pseudónimos en vez de nombres e identificadores verdaderos, la modificación intencional de la información ya sea agregándole ruido o bien reportando una conjunto de posibilidades en vez de la posición real.

Estas posibles técnicas para aumentar el grado de privacidad en la localización, se logran mediante el uso del ruido geográfico, el cual se suma a las coordenadas de tal manera que ampliemos las regiones de localización. También se logra reportando la posición con regiones en vez de puntos, se asegura que se disminuye la certeza del atacante en el momento de realizar el rastreo. Duckham en [14] muestra como realizar dicha técnica.

En la página 11, se definieron cuatro estrategias para proteger la información confidencial de ser revelada, sin embargo sólo dos consideran al usuario como responsable de su propio nivel de privacidad, estas dos maneras son: el anonimato y ofuscación.

El anonimato es una defensa simple contra algún atacante que, de alguna manera, pueda obtener información acerca de nosotros. Consiste simplemente de nunca proporcionar información verdadera que nos pueda identificar, en vez de eso, se brindan pseudónimos. Sin

embargo, esto no es suficiente ya que, como vimos en la sección anterior, algunos investigadores han demostrado que el no saber de quién se trata es indistinto para la obtención de los patrones de movimiento, y por ende en algún momento podría descubrirse el engaño.

Para mejorar la técnica que consiste del reemplazo del nombre o identificadores del usuario con pseudónimos, Beresford y Stanjano en [3] propusieron la idea de cambiar constantemente de pseudónimo mientras se está dentro de la cobertura. Si cada uno de los usuarios hace esto, se reducen las posibilidades de que un atacante pueda reunir suficiente información de los usuarios y pueda realizar el rastreo de manera adecuada. Sin embargo ellos se dieron cuenta de otro posible punto débil, la información de los usuarios debe ser guardada en servidores potencialmente inseguros. Entonces si el atacante vulnera la seguridad de dicho servidor y obtiene la información que vincula los pseudónimos con los usuarios, destruye esta técnica por completo. La privacidad en este esquema no sólo depende de la habilidad de los usuarios de cambiar sus nombres, sino también de la robustez de la infraestructura que soporta dicho esquema, sin embargo se ha continuado con la investigación dada su simplicidad.

Ya vimos cual es el fundamento del anonimato, entonces surge la pregunta, existe una cantidad de usuarios qué hagan tan confusa la información que ya no puedan ser recuperados de manera adecuada los patrones de movimiento. Gruteser y Grunwald en [15] introdujeron el concepto del K-anonimato, este concepto da respuesta a nuestra pregunta. El K-anonimato se refiere a que si una persona reporta, en vez de su posición, la posición de una región en la que se encuentra incluidos k-1 usuarios, entonces dicha persona no puede ser diferenciada dentro del grupo de los k usuarios, por lo que el atacante no sabrá cuál de los k usuarios reportó su posición. Esta ambigüedad también incluye por supuesto, las marcas de tiempo en los registros así como las posiciones.

Sin embargo Betinni en [16], demostró que el K-anonimato puede no ser suficiente para proteger a una persona. Si el atacante se concentra en observar el patrón de la información que cada usuario consulta, o bien de los lugares por los cuales consulta en los servicios basados en la localización, esto podría diferenciar a los usuarios. Podemos entonces observar como la principal deficiencia del anonimato radica en la infraestructura. Esta nueva información puede ser usada para vincular a los usuarios dentro del grupo de K y obtener un patrón identificable. Así mismo Betinni introdujo el concepto del K-anonimato histórico, el cual trata de recuperar la incertidumbre, al inyectar ambigüedad en la información que se consulta.

Este concepto del K-anonimato permitió a Beresford y Stanjano en [3] introducir el concepto de las zonas mixtas. Esencialmente este concepto lo podemos describir de la siguiente manera: los usuarios no transmitirán información de su posición o bien, reportarán su posición cuando se encuentren dentro de zonas especiales llamadas zonas de aplicación. En dichas zonas de aplicación estarán disponibles los servicios basados en la localización y ahí se llevaría acabo el intercambio de pseudónimos. Estas zonas de aplicación podrían ser bancos, aeropuertos, restaurantes, etcétera. Para completar el mecanismo de las zonas mixtas, los pseudónimos se reasignarían al salir de las zonas de aplicación, lo que combinaría dichos pseudónimos entre los usuarios. Los atacantes no sabrían a quien fue asignado cual pseudónimo lo que repercutiría en una mezcla en los patrones de movimiento.

Siguiendo por esta línea de investigación, otra de las maneras de confundir a los atacantes es enviar múltiples peticiones de información, entre las cuales una es la información que se desea obtener y las demás son falsas. Así los servicios basados en la localización responderían a todas ellas y el usuario sólo debe elegir entre las respuestas aquella que le interesa. Se puede notar a simple vista que tipo de debilidad trata de mitigar dicha propuesta, es decir, las debilidades de la infraestructura que implementa el esquema del anonimato. Sin embargo esto genera una sobre carga del tráfico en la red sin tomar en cuenta, cuantas solicitudes hacen falta para disminuir las posibilidades de la localización.

Por otro lado tenemos la ofuscación, este concepto se refiere a la modificación de la información acerca de la localización que se reporta a los servicios. Duckham y Kulik en [17] definen este concepto de la ofuscación mediante la introducción de dos nuevos conceptos, la imprecisión y la inexactitud. La segunda se refiere a proporcionar una medida diferente de la real, en este caso de la posición real, mientras que la imprecisión se refiere a reportar varias posibilidades en vez de una sola posición.

Con respecto a este trabajo Krumm en [18], demostró que la cantidad de ruido aditivo que debe agregarse a la información para disminuir las posibilidades de ser encontrado, es considerablemente alta, usando claro el algoritmo que identifica identidades vía la información GPS.

Gruteser y Hoh en [11], demostraron que existe una vulnerabilidad en el algoritmo de rastreo por objetivo múltiple, ya que al aplicarlo para obtener patrones de movimiento sobre los datos GPS revueltos, notaron que el algoritmo tiene problemas en decir, cuando dos usuarios pasan muy cerca, si de verdad pasaron muy cerca o si se cruzaron. Santosh en [23] explota esta carterística para generar un esquema que confunda al algoritmo de rastreo y que este, no pueda distinguir adecuadamente las trayectorias.

Una manera de mejorar la privacidad tiene que ver con el tiempo que hay entre los reportes de la posición, Hoh en [9] demuestra como disminuye la efectividad del ataque, mientras se aumenta el periodo de tiempo en el que los usuarios reportan su posición. Sin embargo en [20], los autores hacen una revisión de como afecta la introducción de un periodo de silencio entre cambio y cambio de pseudónimos, es decir, cada que termina una sesión de comunicación se realiza un cambio de dirección MAC y de dirección IP para reducir así la precisión de la información.

En [21] también se lleva a cabo un estudio sobre el impacto del cambio dinámico de identificadores, en este caso de las direcciones MAC. En este artículo se propone un esquema en el que cada nodo o usuario entra al sistema por primera vez con una dirección MAC pública determinada, el sistema entonces le asigna una nueva MAC que se encuentra dentro de una lista. Permanece así hasta que llegue el momento de cambiarlas. En este periodo de tiempo cada uno de los nodos de la red le cederá a otro nodo su dirección MAC y él recibirá otra de alguno de los nodos compañeros. Esta medida en conjunto con un periodo de silencio entre cada una de las operaciones permite aumentar la protección en la privacidad de la localización de los usuarios.

Por otra parte, las fuentes de fuga de la información que permite a los atacantes realizar el rastreo son cinco: tiempo, posición, ID del nodo emisor, ID del nodo receptor y contenido de la información. Mientras el contenido puede ser protegido con el uso de encripción y asegurar los servidores que administran la información analizaremos como proteger las otras.

Un enfoque novedoso es el de proteger la posición del usuario mediante la creación de dummies (Nodos falsos), estos dummies tendrán la característica de ser copias del nodo real y tratar de no comprometer la posición del usuario o nodo real, al respecto todavía hay mucho trabajo por hacer, sin embargo en [22], se muestran dos técnicas de creación de las trayectorias. El comportamiento de los dummies, la complejidad de la creación del dummie y hacer posible su éxito, radica en el tiempo de observación del atacante, ya que si este es muy largo, el dummie terminará por exponerse y exponer así la posición real del nodo. En este artículo se describen dos técnicas las cuales se enfocan en crear los movimientos de los dummies mediante patrones, uno con patrón aleatorio y otro con patrón rotativo. En el primero sólo se selecciona el punto inicial y final del movimiento del nodo dummie y algoritmo creará los movimientos aleatoriamente para alcanzar el punto final, mientras que el segundo hace uso de intersecciones con la ruta real del nodo para definir una serie de movimientos realizados por el dummie, sin embargo se observa que esta técnica continúa siendo sensible al tiempo de observación.

2.3. Resumen del capítulo

En el presente capítulo, hicimos una revisión del estado del arte en técnicas de localización y antilocalización, donde resaltamos las principales características así como los objetivos de las mismas. Si bien no se enumeraron absolutamente todos los trabajos al respecto, sí mencionamos los trabajos que tuvieron particular relevancia en nuestra investigación. En la sección de anti-localización pudimos notar las investigaciones que demuestran que el uso de pseudónimos o inclusive el intercambio de identificadores no basta para obtener un grado de privacidad. Es necesario el uso de una estraegia conjunta para obtener mejores resultados. Como lo muestran los trabajos [11], [9], [20] y [22].

Capítulo 3

Desarrollo del algorítmo de estimación de la trayectoria

3.1. Consideraciones para el algoritmo de estimación de la trayectoria

En el capítulo anterior expusimos algunas de las técnicas y estrategias para conseguir la localización (estimación de la trayectoria). Corresponde al presente capítulo el desarrollo de nuestro algoritmo de estimación de la trayectoria, esto con el fin de poder establecer parámetros de comparación para las diferentes técnicas de ofuscación.

Como vimos en el capítulo anterior, existen varias estrategias de localización, pero podemos englobarlas en dos grupos basicamente, las que usan la infraestructura de comunicaciones como el sistema global para las comunicaciones móviles (Global System for Mobile Communications) (GSM) o Wi-Fi, o bien las que usan una infraestructura especializada para la localización, el ejemplo más utilizado para este grupo sería obviamente el GPS. En esta última categoría también podemos encontrar los sensores como por ejemplo sensores de prescencia, sensores Active Bat, etcétera.

Para esta tesis trabajaremos con un algoritmo de estimación de la trayectoria que se encuentra en el primer grupo, es decir, haremos uso de la información propia de la infraestructura destinada para las comunicaciones para recabar información y llevar a cabo la estimación de la trayectoria.

Una vez definido el esquema, resulta necesario especificar con qué mecanismo se llevará a cabo la estimación de la trayectoria. Dicho mecanismo deberá contar con un parámetro que sea común a todos los elementos de la red, esto con el fin de lograr condiciones iguales para todos los nodos de la red. Sabemos bien que estos elementos comunes entre los nodos se encuentran asociados a la tarjeta de comunicaciones. Haciendo referencia a modelos teóricos, podríamos decir que haremos uso de las primeras capas de los modelos de referencia del sistema abierto de interconección (Open System Interconnection) (OSI) y Protocolo de control de transmisión (Transmission Control Protocol) (TCP)/Protocolo de Internet (Internet Protocol) (IP). Los parámetros que debemos considerar más útiles en términos de localización se encuentran

propiamente alojados en capa física del modelo OSI (PHY). Debemos elegir un parámetro el cual se encuentre relacionado con la distancia y así establecer con base en el, el mecanismo de estimación de la trayectoria.

Dentro de los parámetros mencionados previamente, debemos hacer una selección de acuerdo con el esquema de estimación de la trayectoria y el área sobre la que se desplegará el sistema. Primero podríamos considerar el parámetro tiempo de llegada (Time of Arrival) (ToA) o bien diferencial del tiempo de llegada (Time Diference of Arrival) (TDoA) que permiten medir el tiempo que debe transcurrir para que las ondas se propaguen por el medio inalámbrico, desde el emisor hasta el receptor. El parámetro ToA es directamente proporcional a la distancia entre el emisor y el receptor e inversamente proporcional a la velocidad de propagación de las ondas electromagnéticas en el espacio libre. Como podemos ver, para distancias relativamente pequeñas, como las que podemos encontrar en una WLAN, alrededor de 100 m, los tiempos involucrados para determinar las distancias en este esquema de estimación de la trayectoria corresponderían a valores menores a microsegundos, lo que haría necesario el uso de relojes muy precisos tanto en el emisor como en el receptor. Dichos relojes no se encuentran disponibles en dispositivos de radio de WLAN convencionales.

Otro parámetro que podría utilizarse es el de ángulo de llegada (Angel fo Arrival) (AoA). Este parámetro es ideal para determinar trayectorias en redes celulares, en las cuales las radiobases cuentan con antenas sectoriales. Sin embargo, para este trabajo consideramos que tanto las radio bases como el nodo requerirán hacer uso del esquema de estimación de la trayectoria. Utilizar AoA dejaría en clara desventaja a los nodos móviles dado que, por cuestiones de costo e implementación, no cuentan con este tipo de antenas.

Uno de los parámetros más sencillos de estimar y que no cuenta con los limitantes previos es el uso del parámetro intensidad de la señal (Siganl Strenght) (SS), el cual es dependiente de la distancia y puede ser determinado ya sea por las radio bases o bien por los nodos móviles.

La atenuación de las señales en el medio inalámbrico puede ser asociada a la distancia entre los dos dispositivos. Sin embargo, para establecer esta relación debemos contar con un modelo de propagación adecuado, ya que éste establecerá la relación entre potencias tanto transmitida como recibida y la distancia entre el nodo emisor y el receptor.

Por sencillez en la implementación tomaremos como referencia una red WLAN. Con dicha red podremos llevar a cabo las simulaciones correspondientes que muestren el comportamiento del esquema de estimación de la trayectoria desarrollado en esta tesis.

En [24] se muestra el algoritmo Guiding Users in Distributed Enviroments (GUIDE), dicho algoritmo se utiliza para guiar un nodo móvil en WLAN hacia otro nodo móvil. En dicho artículo se muestra el modelo de propagación utilizado comúnmente en WLAN, corresponde al modelo de dos rayos, el cual se muestra a continuación:

$$P_{rx}(d) = \frac{P_{tx} * G_{tx} * G_{rx} * H_{tx}^2 * H_{rx}^2}{d^4 * L} \tag{3.1}$$

donde $G_{tx}, G_{rx}, H_{tx}, H_{rx}$ corresponden a las ganancias y a las alturas de las antenas del transmisor y el receptor, respectivamente; d es la distancia y L es un factor de atenuación del enlace (Link Loss Factor). Si suponemos antenas omnidireccionales, dadas las razones expuestas previamente, podemos tomar las ganancias de las dos antenas igual a 1. De dicho modelo podemos observar claramente una expresón que relaciona potencias, tanto de transmisión como de recepción, con la distancia entre los equipos. Para nuestro caso podremos obtener las distancias entre el dispositivo móvil y alguna radio base determinada o viceversa. En particular, las radio bases corresponden a los puntos de accesso (Access Points) (AP)s de WLAN.

Una vez obtenida dicha relación, podemos entonces establecer un esquema que nos permita conocer la distancia entre un nodo móvil y el punto de referencia (un AP). Una vez conocidas dichas distancias debemos relacionarlas entre sí para obtener la posición de nuestro nodo móvil. De acuerdo con el artículo [23], uno de los esquemas más ampliamente usados para llevar a cabo la determinación de la posición corresponde al esquema de multilateración. En este esquema se encuentra la posición de un punto usando como referencia la distancia entre ese punto desconocido y tres o cuatro puntos no colinieales de posiciones conocidas. En nuestro caso esos tres o cuatro puntos conocidos corresponden a las posiciones de los APs. Es posible asumir esto ya que en una red WLAN los APs tienen posiciones físicas determinadas y conocidas por los administradores de dicha infraestructura y, por lo tanto, permitirían así determinar la posición correspondiente al nodo móvil unicamente con los equipos que forman la infraestructura de comunicaciones.

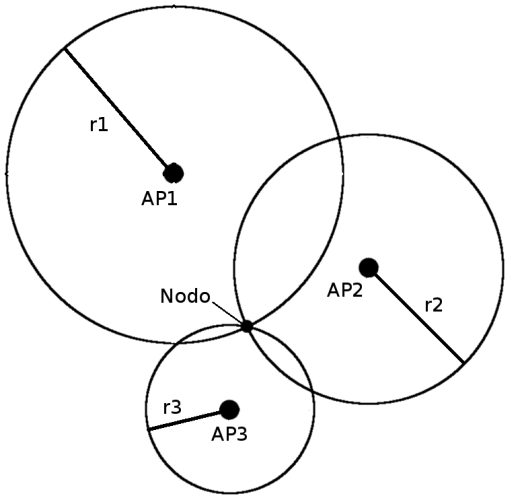


Figura 3.1: Trilateración ideal.

En la Figura 3.1 se ilustra la manera en la que un conjunto de APs llevan a cabo la estimación de la trayectoria de un nodo móvil. Cabe señalar que esta es una primera aproximación la cual considera una situación ideal, en la cual no se toman en cuenta condiciones del terreno,

ni muchos otros factores que afectan la propagación de las señales en un medio inalámbrico.

Otro aspecto importante a resaltar es que este esquema reduce el problema de la estimación de la trayectoria a un problema geométrico. Si tomáramos en cuenta los errores en la medición, podríamos entonces visualizar la estimación de la trayectoria del nodo móvil ya no como un punto, sino como una zona. Esta zona corresponde a la región sombreada que se muestra en la Figura 3.2.

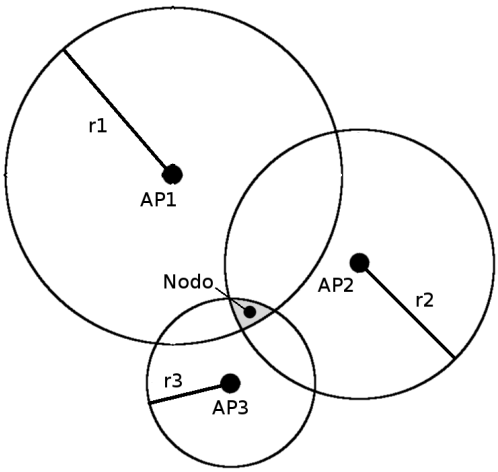


Figura 3.2: Trilateración con error.

Si continuamos con la idea de la sintetización geométrica entonces podemos evaluar varios casos de la estimación de la trayectoria como simples situaciones geométricas. Por ejemplo, si tomáramos en cuenta un solo AP como referencia y un solo nodo aproximándose, el rastreo correspondería a un conjunto de puntos que forman una circunferencia. Esto debido a que si consideramos que sólo se puede determinar la distancia entre el nodo y el AP, y dado que no se cuenta con más información debemos asumir como posibles todas las soluciones de dicho resultado. Así mientras el nodo se desplaza todos los puntos resultantes del mapa corresponderían a circunferencias con centro en el AP, que variarían su diámetro mientras en nodo sigue una trayectoria dada.

Es importante tener presente que si bien una circunferencia matemáticamente está formada por una infinidad de puntos, en la realidad no podemos considerarla así. La naturaleza de las posiciones geográficas es discreta por lo que la consideraremos como una malla de puntos. Tomando en cuenta lo anterior, supongamos que tenemos como resultado un número n de puntos que pertenecen a una circunferencia dada. Uno de esos puntos se encuentra próximo a la posición real del nodo móvil y hay n-1 puntos que son posiciones falsas, pero el AP no tendría información suficiente como para discriminar dichos puntos y aproximar mejor la posición del nodo móvil.

Otro caso para ser analizado es cuando se está al alcance de dos APs simultaneamente. En este caso la solución geométrica de dos circunferencias que se intersecan, son dos puntos. Esto reduce por mucho las posibilidades de ser confundido entre algunos de los posibles puntos estimados por el algoritmo, si tomamos en cuenta el caso anterior cambia de haber n-1 puntos falsos a haber sólo un punto falso.

Por último tenemos el caso de tener tres APs al alcance simultáneamente, en dicho caso y suponiendo un caso ideal en el modelo de propagación, dicho rastreo sólo tiene como posibilidad un solo punto, el correspondiente al nodo real. Es decir, en este caso no se genera ningún punto falso debido a lo siguiente: si cuando se tienen dos APs se tienen sólo dos posibilidades, el alcance del tercer AP intersecta con sólo uno de los dos puntos definidos anteriormente, por tanto, el otro punto posible se descartará. Este caso es el más complicado dado que no hay posibilidades de confundir a los APs, inclusive se puede demostrar que más de tres APs con su cobertura intersectada producen el mismo resultado, en la Figura 3.3, se muestran las imágenes que ilustran los tres casos anteriores.

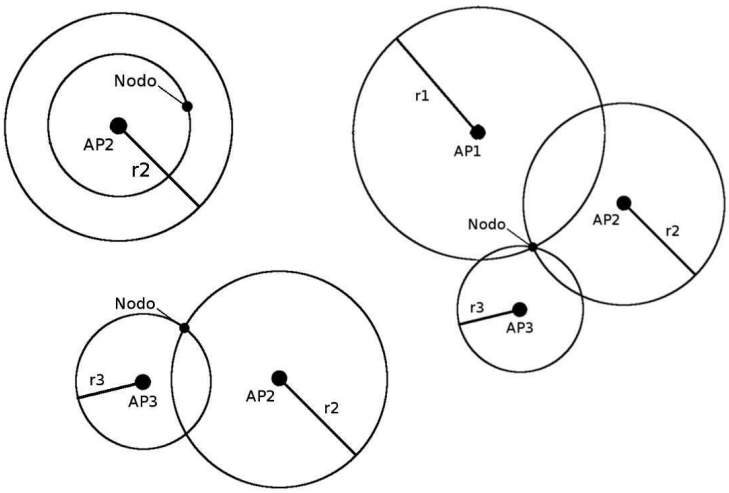


Figura 3.3: Tres casos de trilateración.

Si suponemos un escenario donde tengamos varios puntos de acceso, un nodo que se desplaza en una trayectoria dada y aplicamos todos los conceptos anteriores para tratar de identificar cual fue su trayectoria, obtendríamos lo que se muestra en la Figura 3.4.

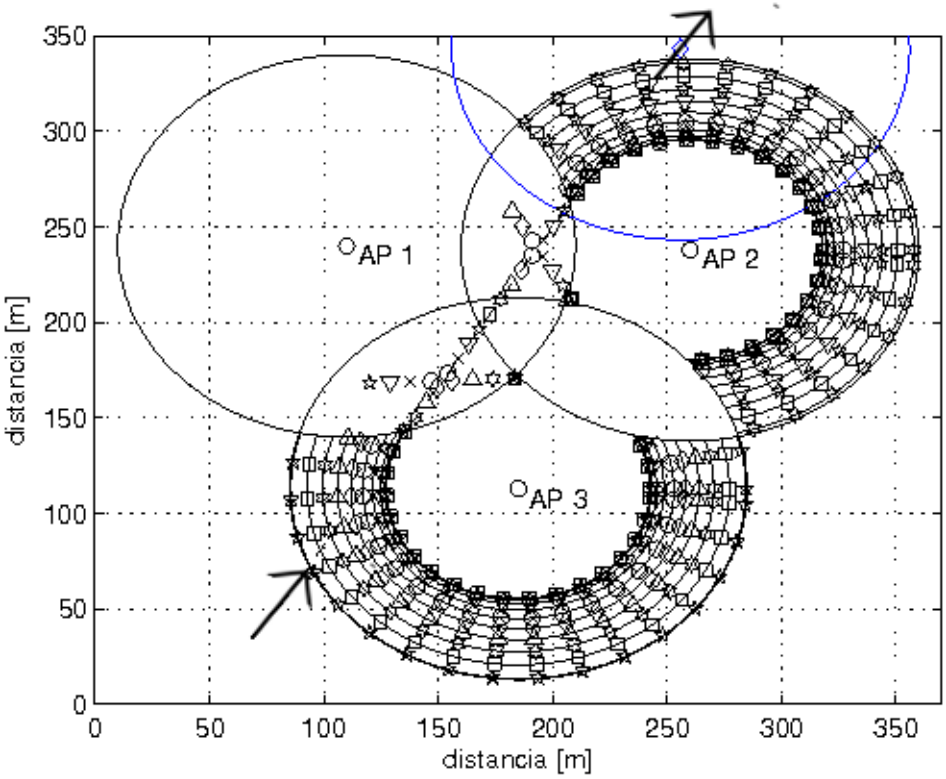


Figura 3.4: Mapa generado por el algoritmo de estimación de la trayectoria.

En la Figura 3.4 se puede ver como al ir pasando por las diferentes regiones de la cobertura entre los tres APs se van formando patrones diferentes de huellas. También podemos observar como el número de puntos falsos disminuye mientras aumenta el número de APs que comparten la cobertura. En el caso de la imagen al principio tenemos n=19 con 18 puntos falsos, después pasa a n=2 con 1 punto falso y en el centro de la imagen a n=1, con cero puntos falsos. Estos últimos puntos le dan una ventaja al atacante ya que son puntos con una certeza del 100 %.

En la Figura 3.4, podemos observar que para cada conjunto de puntos por los cuales pasa el nodo móvil a lo largo de trayectoria, son graficados con pequeñas figuras. Dichas figuras cambian entre conjuntos, esto es porque especifican que fueron hechas en intervalos de tiempo diferentes y cada una pertenece a un conjunto de puntos, los cuales tienen como característica haber sido registrados en el mismo tiempo. En una primera aproximación, cada una de las posiciones de cada conjunto son equiprobables con respecto a los demás puntos del mismo conjunto. El mapa previo es obtenido sólo con la información de la potencia, es por esto que los puntos se consideran indistinguibles entre ellos. Es necesaria la obtención de más información para poder determinar con mayor precisión la ubicación del nodo real, dicha información nos permitiría asignar posiciones más probables a cada punto del conjunto de

posibles posiciones.

Cabe señalar que el resultado mostrado anteriormente con respecto algoritmo de estimación de la trayectoria hace un conjunto de suposiciones para poder mostrar los puntos como los vemos en la Figura 3.4. Esto se puede notar a simple vista, si observamos que la estimación de la trayectoria generada para cuando el nodo móvil se encuentra sólo al alcance de un AP no son circunferencias sino segmentos de una circunferencia, esto debido a que suponemos en la implementación una colaboración entre los APs, lo que da como resultado que se eliminen ciertos puntos del total del conjunto por intervalo de tiempo. Esta es una suposición válida ya que dentro de la infraestrutura de comunicaciones los APs deben mantenerse comunicados entre si con fines de procesos tales como los handovers. En este caso, esa comunicación será útil en términos de verificar si el nodo que se está detectando dentro de la cobertura, también se encuentra dentro de la cobertura de un AP vecino. De ser así, la región donde se encuentra el nodo móvil puede ser acotada en espacio. En el caso de la Figura 3.4, mientras el nodo móvil se desplaza sólo dentro de la cobertura del AP1, dicho AP interrogará a sus vecinos, por si ellos también pueden "ver" al nodo móvil, dado que no es el caso, el AP1 elimina como posibilidades a los puntos que se encuentran sobre la circunferencia trazada que se encuentran en la intersección con los APs vecinos. Cuando el nodo real se encuentra verdaderamente en una región de intersección de coberturas, el AP ignorará todas las posibilidades y sólo tomará en cuenta las dos que son soluciones a la intersección de las circunferencias estimadas por el algoritmo.

Dichas suposiciones nos permiten aportar un poco más de información al proceso de estimación de la trayectoria, lo cual permite reajustar las probabilidades de que los puntos se encuentren en alguna posición dentro de nuestro conjunto de puntos. Sin embargo, las consideraremos etapas separadas del proceso de estimación de la trayectoria, ya que dada la carga de operaciones que debe realizar un AP y su limitada capacidad de procesamiento y almacenamiento, dichos elementos de la red no podrían llevar a cabo la estimación de la trayectoria de todos los nodos móviles dentro de su cobertura, los APs se limitarían a reportar a un tercero la información obtenida.

Si bien el mapa mostrado en la Figura 3.4, nos permite tener una idea poco aproximada de la trayectoria descrita por el nodo móvil, aún no podemos considerarlo un algoritmo de estimación de la trayectoria terminado ya que sigue teniendo muchas posibilidades y la trayectoria del nodo sigue siendo muy vaga. Usaremos esta primera aproximación para poder completar el algoritmo mediante una etapa de posprocesamiento que termine de refinar la trayectoria.

3.2. Etapa de posprocesamiento

Si partimos del mapa proporcionado por la etapa previamente descrita, el cual se muestra en la Figura 3.4, debemos notar que el nivel de información en cada conjunto de puntos es diferente. Como vimos en la sección 3.1 (consideraciones para el algoritmo de estimación de la

trayectoria), los conjuntos de puntos que brindan mayor información, para poder establecer un criterio de discriminación, son los que contienen uno o dos puntos estimados por el algoritmo. En otras palabras, las posibilidades que la posición de un nodo móvil se encuentre dentro de la cobertura compartida entre dos y tres APs.

Si nos concentramos en dichos conjuntos de puntos podremos observar un patrón predecible de puntos, la Figura 3.5 nos permite observar con mayor detalle dicho patrón.

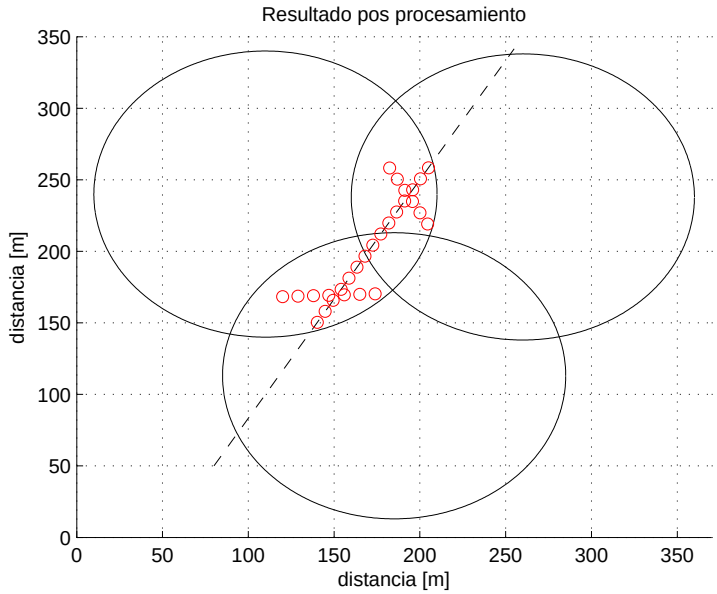


Figura 3.5: Puntos de los conjuntos de tamaño igual a 1 y 2.

En la Figura 3.5 podemos ver a simple vista la trayectoria descrita por el nodo móvil ya que si bien, en las áreas de intersección de sólo 2 APs se muestran dos posibilidades, sólo un subconjunto de ellas concuerda con el patrón dejado cuando el nodo móvil cruzó la zona de cobertura compartida por los 3 APs, ¿pero cómo encontrar dicho patrón de manera automática?

Para establecer un criterio que nos permita discriminar entre las dos posibilidades que se generan en la zona de cobertura compartida entre dos APs, recurriremos a la definición analítica de una recta. Debemos de tener presente que para generar una recta se debe contar con dos puntos, o bien con un punto y un vector director. Dicho vector tiene las característica de ser unitario y estar asociado a cualquier punto de la recta. Por lo que si encontramos la manera de asociar vectores unitarios a cada punto, podríamos diferenciar que puntos pertenecen a la trayectoria seguida por el nodo móvil.

Para poder llevar a cabo el procedimiento anterior, es fundamental asignar de manera adecuada los vectores unitarios a cada punto del mapa, para hacer eso, es preciso seleccionar un punto de referencia. Para obtener dicho punto de referencia, ordenamos primero los conjuntos

de puntos cronológicamente. En el primer conjunto habrá dos puntos, como mencionamos en la sección anterior uno de esos puntos es falso y el otro es verdadero. En otras palabras uno pertenece a una recta y el otro a una recta diferente. Una de estas dos rectas es igual a la recta descrita por el movimiento del nodo móvil. Si partimos de dichos puntos como referencia para obtener vectores unitarios asociados a todos los demás puntos de los conjuntos subsecuentes, el resultado se muestra en la Figura 3.6.

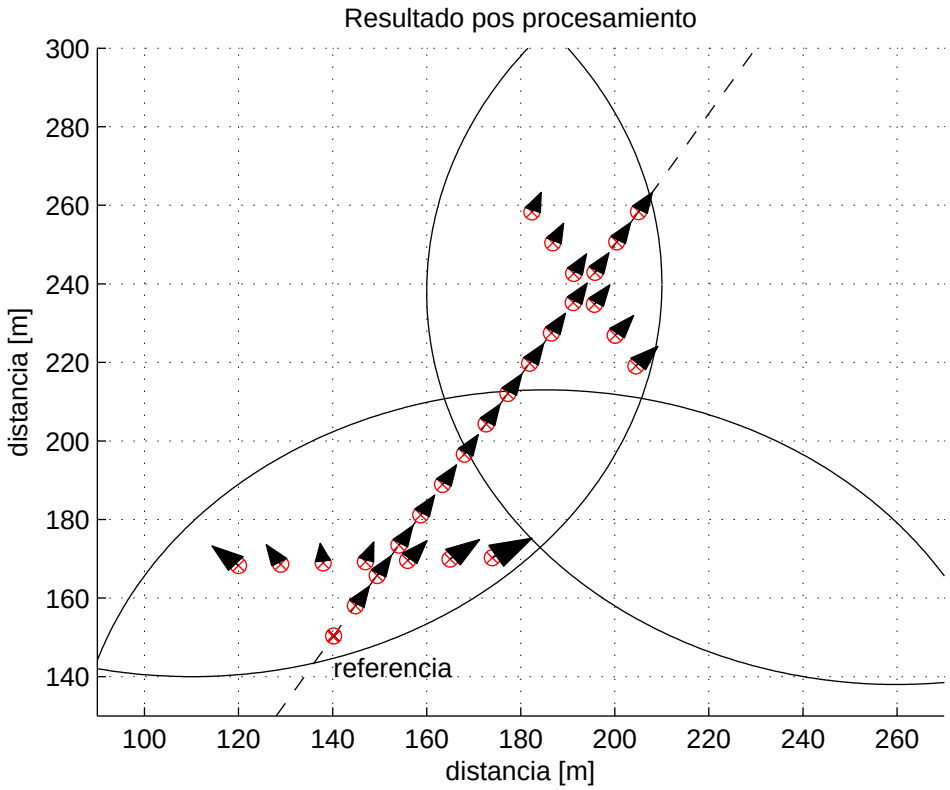


Figura 3.6: Mapa de las ubicaciones geográficas con vectores asociados.

Habiendo asociado el punto de referencia indicado en la Figura 3.6, obtenemos los vectores unitarios asociados a los puntos de los conjuntos sucesivos. De la Figura 3.6 podemos observar que los vectores unitarios encontrados, forman parte de distintas rectas. Algunos puntos tienen asociado el mismo vector unitario. Entonces, sólo bastará con contar el número de vectores unitarios parecidos entre sí, para encontrar la trayectoria seguida por el nodo móvil. Por esto, sólo consideraremos los puntos que están asociados al vector unitario con mayor número de coincidencias. El resultado se muestra en la Figura 3.7.

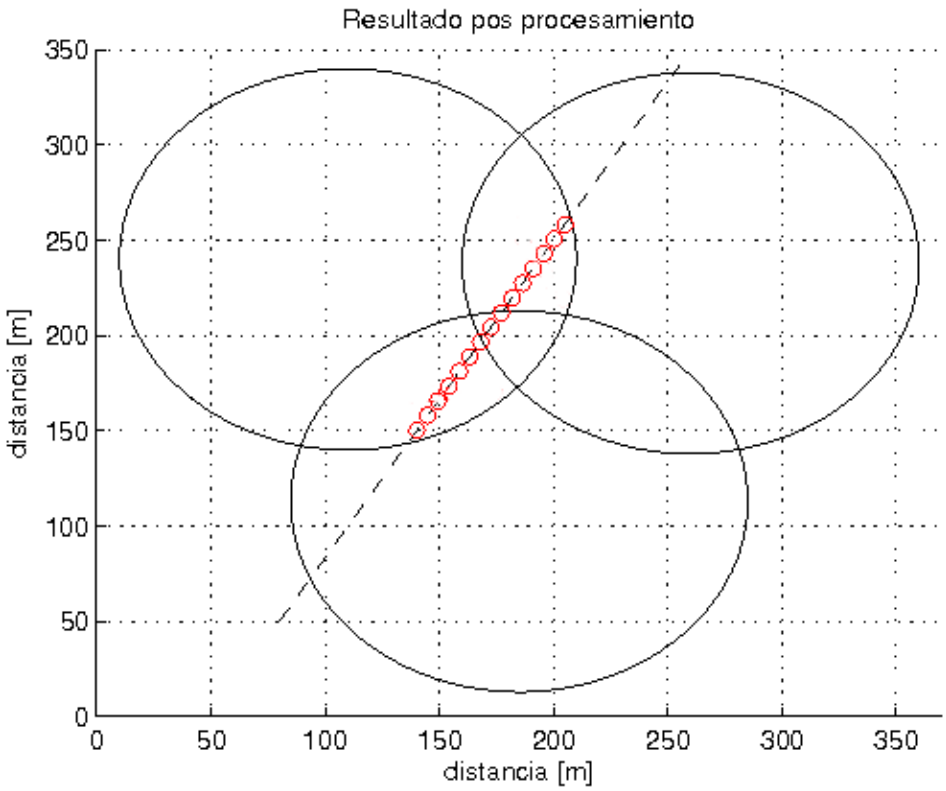


Figura 3.7: Mapa de las ubicaciones geográficas determinadas como verdaderas por el algoritmo.

En la Figura 3.7 se observa que el objetivo propuesto fue logrado. El patrón distinguible de puntos puede obtenerse con el procedimiento descrito, consiguiendo así la estimación de la trayectoria del nodo móvil. Con base en ese conjunto reducido de puntos podemos obtener una regresión polinomial, con la cual discriminar los conjuntos de puntos sucesivos.

Por último, sólo queda considerar dos últimos casos. El primero se refiere a cuando un solo nodo móvil se aproxima al área de cobertura de un único AP. En dicho caso, no hay realmente mucha información de la cual echar mano, ya que sólo contamos con conjuntos de n puntos y la cantidad de rectas que pueden trazarse entre dichos puntos es muy grande. Sin ningún otro elemento que brinde información resultaría inútil tratar de determinar la recta que describa la trayectoria. Recurriremos a una regresión lineal para establecer un resultado en dicho caso. El segundo caso se refiere a cuando el nodo móvil sólo cruza una de las áreas de cobertura de dos APs, como se muestra en la Figura 3.8.

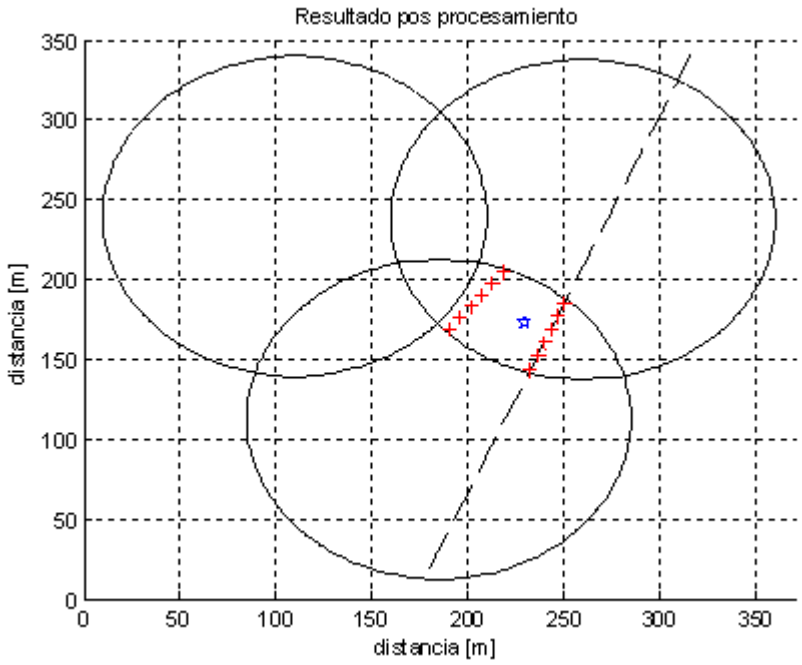


Figura 3.8: Mapa de ubicaciones para caso especial de dos APs.

En la Figura 3.8 podemos observar que el procedimiento usado para determinar la trayectoria presenta un caso especial. Podemos observar que hay dos rectas descritas por los puntos de los conjuntos, si realizáramos el conteo de vectores unitarios tendríamos dos vectores unitarios solamente, y el número de puntos que estan asociados a dichos vectores son los mismos. Pero sólo uno de dichos conjuntos de puntos es correcto y el otro es el generado por las soluciones falsas. Dado que no hay información suficiente para tomar una decisión, recurriremos a un criterio nuevo. Cuando ocurra este caso, obtendremos el promedio de las coordenadas de todos los puntos en el mapa y por medio de dicho punto seleccionaremos el conjunto de puntos que se encuentra más próximo. Sin embargo esta solución introduce errores al algoritmo, esto considerando que bajo ciertas condiciones muy particulares, el algoritmo escoge el conjunto de puntos equivocada.

Considero pertinente hacer un breve resumen de las etapas en la que dividimos nuestro algoritmo de estimación de la trayectoria para mantenerlo en mente:

1. Los APs reportan mediciones de potencia y tiempo asociadas a un nodo.
2. Se lleva a cabo una sintetización de puntos basada en las características geométricas, es decir si se encuentra en una región de intersección.
3. Se determina la ruta más probable tomando en cuenta el algoritmo que asocia vectores unitarios al conjunto de puntos.

4. Por último se hace una reducción de los últimos conjuntos de puntos, tomando en cuenta su proximidad a la recta obtenida por el paso anterior.

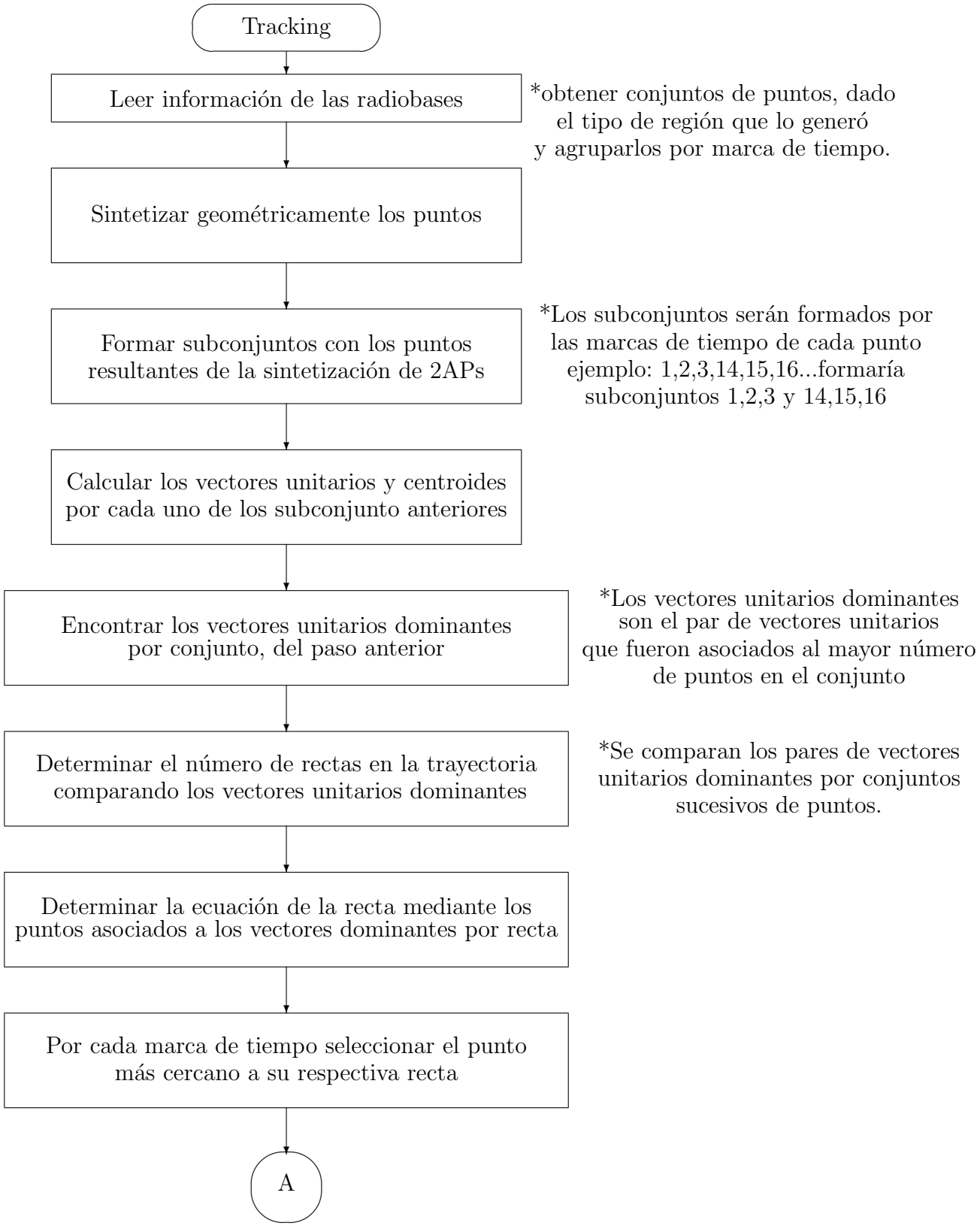
La etapa de posprocesamiento previamente descrita, consiste de la primera aproximación que desarrollamos para esta tesis. Sin embargo, mientras avanzamos en el desarrollo de la misma nos dimos cuenta de algunas de sus fallas. Por ejemplo, este enfoque sólo funciona bien cuando se tiene una trayectoria lineal. En una etapa posterior del desarrollo, incluimos las rutas generas por el modelo Random Waypoint (RWP), dicho modelo permite generar trayectorias más realistas para nuestros nodos móviles que incluye cambio de trayectorias, sin embargo introdujo una nueva complejidad. Como ya sabemos, dicho modelo construye una ruta desde un punto inicial hasta un punto final con segmentos de rectas calculadas de manera aleatoria, esto implica múltiples segmentos de recta en una sola ruta.

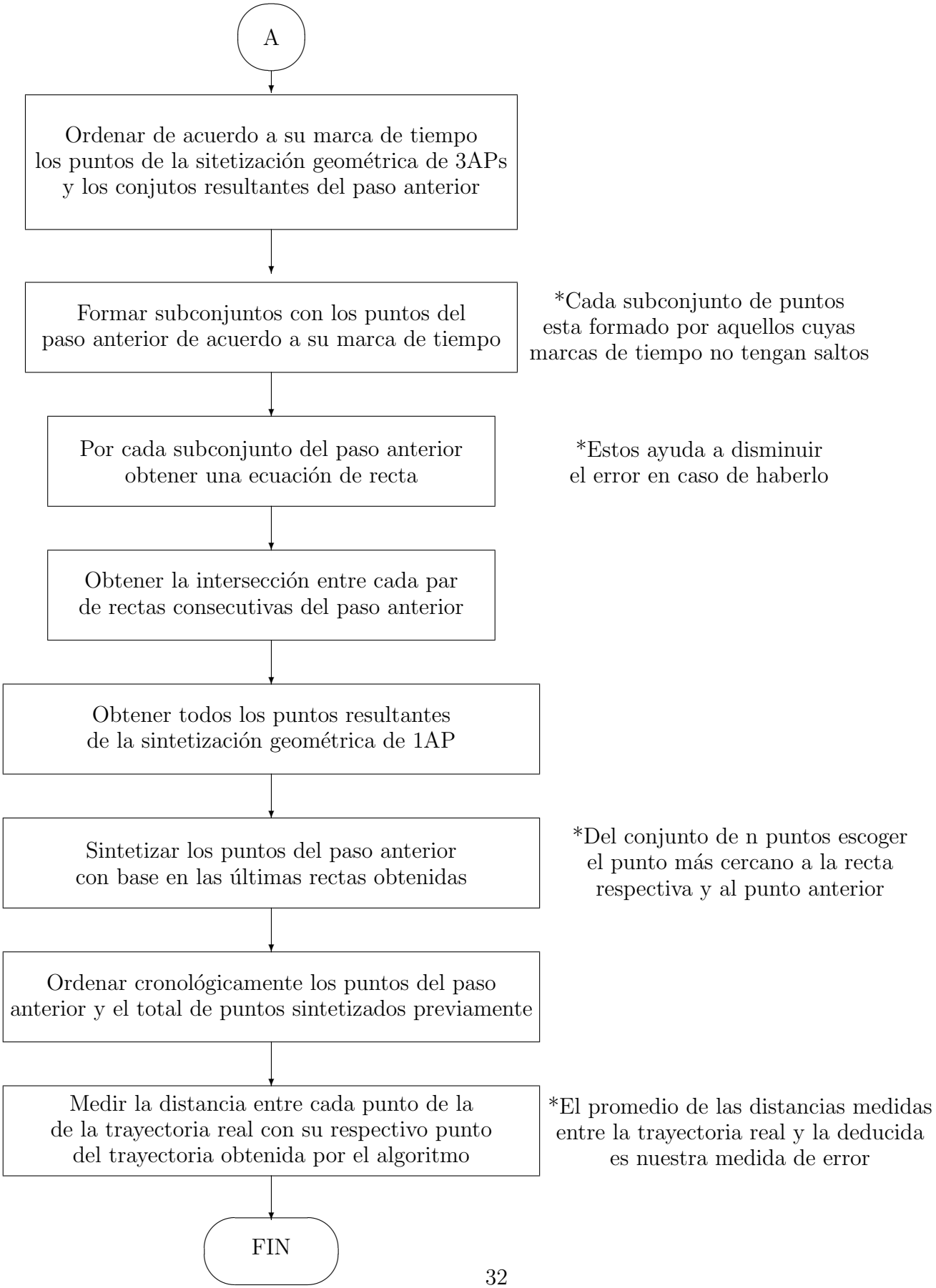
Para poder superar esta complejidad modificamos la aplicación del algoritmo anterior, aunque sigue utilizando los mismos principios básicos, tiene algunos elementos extra que le permitieron trabajar con trayectorias más complejas.

El primer reto fue, dado el conjunto de puntos en un mapa como el resultado de la etapa previa, poder diferenciar qué puntos pertenecen a una recta y cuáles a la siguiente recta. Propusimos un algoritmoo muy sencillo para lograr superar esta complicación. Si consideramos a las regiones de intersección de dos APs como referencia, y encontramos los dos vectores unitarios predominantes en cada conjunto. Podríamos comparar dichos vectores unitarios con el par de vectores unitarios del siguiente conjunto de puntos, en donde podríamos encontrar sólo dos resultados. Uno de los resultados es que uno de los vectores unitarios del primer par de vectores se parezca a uno del siguiente par, lo cual implicaria que pertencen a una misma recta, o bien que ninguno de los cuatro vectores se parezca entre sí, lo cual implicaría que ambos conjuntos pertencen a rectas diferentes.

Cabe señalar, dado que hacemos uso de los puntos en las regiones de intersección de dos APs para delimitar las rectas, que si una intersección cae en una región donde se intersequen dos APs, no podremos localizarla y sólo podremos determinar el cambio de recta. Sin embargo sólo se estarían perdiendo una cierta cantidad de puntos, ya que el algoritmo de procesamiento considera y sintetiza los puntos de las demás regiones tomando en cuenta a las otras rectas.

A continuacion explico en un diagrama de flujo el algoritmo completo de estimación de la trayectoria que será implementado en Matlab.


Tracking
Leer información de las radiobases
*obtener conjuntos de puntos, dado el tipo de región que lo generó y agruparlos por marca de tiempo.
Sintetizar geométricamente los puntos
Formar subconjuntos con los puntos resultantes de la sintetización de 2APs
*Los subconjuntos serán formados por las marcas de tiempo de cada punto ejemplo: 1,2,3,14,15,16...formaría subconjuntos 1,2,3 y 14,15,16
Calcular los vectores unitarios y centroides por cada uno de los subconjunto anteriores
Encontrar los vectores unitarios dominantes por conjunto, del paso anterior
*Los vectores unitarios dominantes son el par de vectores unitarios que fueron asociados al mayor número de puntos en el conjunto
Determinar el número de rectas en la trayectoria comparando los vectores unitarios dominantes
*Se comparan los pares de vectores unitarios dominantes por conjuntos sucesivos de puntos.
Determinar la ecuación de la recta mediante los puntos asociados a los vectores dominantes por recta
Por cada marca de tiempo seleccionar el punto más cercano a su respectiva recta
A


A
Ordenar de acuerdo a su marca de tiempo los puntos de la sitetización geométrica de 3APs y los conjutos resultantes del paso anterior
Formar subconjuntos con los puntos del paso anterior de acuerdo a su marca de tiempo
*Cada subconjunto de puntos esta formado por aquellos cuyas marcas de tiempo no tengan saltos
Por cada subconjunto del paso anterior obtener una ecuación de recta
*Estos ayuda a disminuir el error en caso de haberlo
Obtener la intersección entre cada par de rectas consecutivas del paso anterior
Obtener todos los puntos resultantes de la sintetización geométrica de 1AP
Sintetizar los puntos del paso anterior con base en las últimas rectas obtenidas
*Del conjunto de n puntos escoger el punto más cercano a la recta respectiva y al punto anterior
Ordenar cronológicamente los puntos del paso anterior y el total de puntos sintetizados previamente
Medir la distancia entre cada punto de la de la trayectoria real con su respectivo punto del trayectoria obtenida por el algoritmo
*El promedio de las distancias medidas entre la trayectoria real y la deducida es nuestra medida de error
FIN

3.3. Resumen del capítulo

En este capítulo llevamos a cabo la definición del algoritmo de estimación de la trayectoria empleado en la presente tesis. Dicho algoritmo hace uso de las caracterísiticas geométricas de los conjuntos de puntos para poder identificar los movimientos de los nodos móviles que se desplazan dentro de la zona de covertura. Se utilizó un digrama de flujo para explicar más a detalle el funcionamiento del algoritmo de estimación de la trayectoria.

La implementación del algoritmo previamente descrito fue realizada en MatLab, los resultados serán discutidos más adelante en esta tesis. El código fuente de la implementación se encuentrán en el apéndice A.

Capítulo 4

Técnicas de Ofuscación

Una vez definido el algoritmo mediante el cual llevaremos a cabo la estimación de la trayectoria, corresponde al presente capítulo el desarrollo de la propuesta de esta tesis. Se plantea proponer un mecanismo que impida o disminuya la posibilidad de realizar la estimación indeseada de la trayectoria de los nodos móviles. Se implementará y probarán dos técnicas de ofuscación que permitan un mayor grado de privacidad para los nodos dentro de una red. La primer técnica corresponde al control de potencia. Como indica su nombre, el objetivo es variar la potencia de transmisión del nodo móvil de tal manera que los APs no puedan inferir de manera adecuada su posición goegráfica. La segunda técnica consiste en la creación de agentes virtuales variando parámetros tales como: identificadores, direcciones Media Access Control (MAC) e IP y potencia de transmisión. Esta variación podría generar un entorno en el cual una red o agente externo, no sea capaz de identificar la entrada y salida de nodos reales. Esto traería como consecuencia que no se pueda llevar a cabo de manera indeseada: la localización, identificación y seguimiento del usuario móvil.

4.1. Control de potencia

Esta estrategia se refiere a variar la potencia de transmisión de tal manera que la localización sea errada.

¿Cómo funciona esta estrategia? sabemos que el algoritmo de estimación de la trayectoria requiere del modelo de propagación para poder transformar lecturas de potencia en distancias. En la capítulo 3 detallamos dicho modelo. Sin embargo, cabe mencionar que este modelo requiere de la potencia de transmisión así como la potencia recibida para poder determinar la distancia que separa los nodos en cuestión. Si el AP sólo conoce la potencia de recibida, debe asumir que la potencia de transmisión es la potencia que usan todos los nodos por omisión. Sin embargo, si un nodo cambia su potencia de transmisión por una potencia diferente a la definida por el estándar, esto afectaría el valor de distancia estimado por el algoritmo de estimación de la trayectoria, así podría inducirse intensionalmente un error en la estimación de la trayectoria.

¿Pero cómo variar efectivamente dicha potencia? Una de las premisas fundamentales de esta tesis es que tanto el AP como el nodo móvil pudieran llevar a cabo la estimación de la

trayectoria. Si un nodo determina su distancia con respecto a la radiobase, tomando en cuenta las lecturas de potencia provenientes de la misma, podría llevar a cabo el ajuste de potencia necesario. Si conocemos la distancia entre los nodos y definimos que el nodo AP deberia recibir nuestra señal con una determinada potencia de recepción, sólo debemos despejar la potencia de transmisión del modelo de propagación y llevar a cabo la transmisión con dicho valor para terminar con el engaño. Exploraremos dos casos en esta estrategia, uno es el que denominamos potencia mínima y otro que denominamos potencia variable.

Todos los nodos utilizan por omisión la potencia máxima, esta potencia estará determinada por el estándar, en el caso de WLAN corresponde a PtxDefault=100 [mW]

4.1.1. Potencia mínima

El control de potencia mínima consiste en ajustar la potencia del nodo móvil, de tal manera que el nodo que reciba la señal siempre mida la potencia mínima de operación. Esta estrategia, a primera vista, nos permite engañar al nodo receptor, dado que no se puede determinar de manera adecuada la posición real del nodo móvil. Recordemos que uno de los modelos de propagación más utilizado en WLAN es el modelo de propagación de los dos rayos. Para este caso haremos uso de una versión simplificada del modelo mostrado en la ecuación (3.1):

$$P_{rx}(d) = \frac{P_{tx}}{d^4}[W] \tag{4.1}$$

Donde P_{rx} es la potencia recibida en el nodo receptor, P_{tx} es la potencia de transmisión en el nodo emisor y d es la distancia en metros que los separa.

Si un AP recibe $P_{rx} = P_{ApRx}$, y toma en cuenta que $P_{Tx} = P_{TxDefault}$, se puede determinar la distancia estimada del nodo al AP, la cual estaría dada por:

$$d_{nodo-AP} = \sqrt[4]{\frac{P_{TxDefault}}{P_{ApRx}}} \tag{4.2}$$

Una vez conocida dicha distancia, podemos ajustar la potencia de transmisión de tal manera que sólo la potencia mínima alcanze al AP. Para conseguirlo es necesario primero determinar cuál es dicha potencia mínima necesaria para operar. Esta potencia está definida por el estándar, en el caso de WLAN la potencia mínima corresponde a $P_{RxMin} = 1x10^{-6}$ [mW], así $P_{Rx} = P_{RxMin}$ y la ecuación queda de la siguiente manera:

$$P_{Tx} = P_{RxMin} * d^4_{nodo-AP} \tag{4.3}$$

Sin embargo, existen algunos detalles que debemos mencionar de dicha estrategia. Una de las potenciales ventajas de esta estrategia es que al variar la potencia durante una trayectoria de

movimiento, se escoge al AP más próximo como referencia para ajustar la potencia mínima que debe recibir dicho nodo. Esto implica que gran parte del tiempo reducimos la potencia por debajo de la potencia nominal de transmisión, lo cual repercute positivamente en la energía necesaria para la operación del nodo. En la Figura 4.1 se muestra la potencia de transmisión usada durante una trayectoria determinada, donde un solo nodo móvil cruza por una zona de cobertura de 3 APs realizando control de potencia mínima.

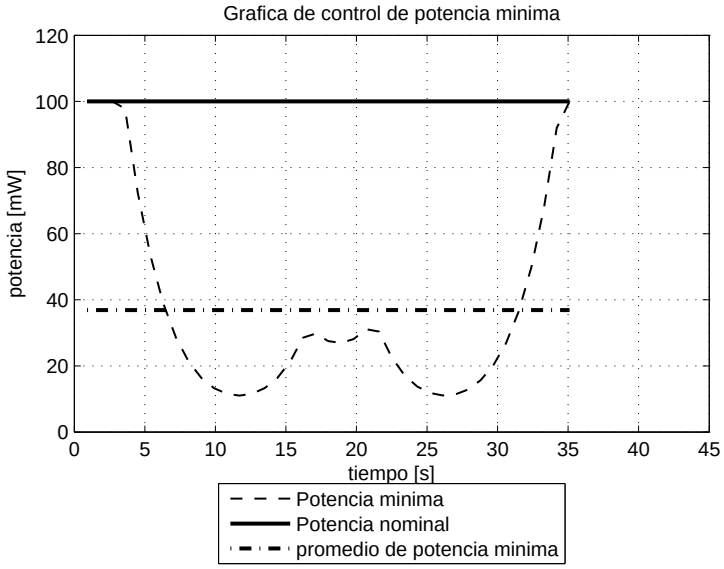


Figura 4.1: Gráfica de la comparación de potencias en la estrategia control de potencia mínima.

En la Figura 4.1, se puede observar como la potencia promedio utilizada, resulta ser menor a la mitad de la potencia necesaria bajo condiciones de operación normales. Sin embargo, la ventaja más considerable es la simulación de un movimiento aparente. Como sabemos, al ajustar la potencia al mínimo necesario, dicho nodo aparenterá a los APs que su posición es la más alejada posible de los mismos. Además, dado que suponemos una colaboración inter AP, esto determina una región específica de localización, la cual es independiente del movimiento que el móvil este realizando. Dicha región se muestra a continuación en la Figura 4.2.

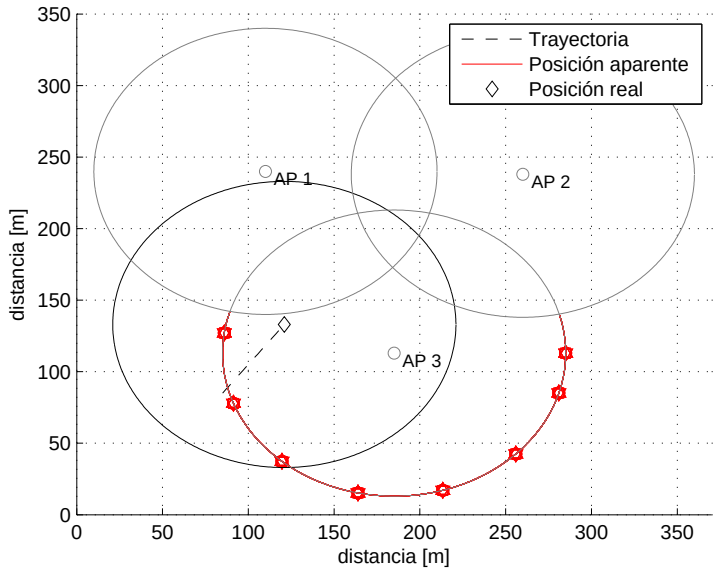


Figura 4.2: Mapa de ubicación generado por el algoritmo cuando esta activo el control de potencia mínima.

Como podemos observar en la Figura 4.2, a pesar del movimiento real del nodo móvil, el movimiento aparente (para las radiobases y el sistema de estimación de la trayectoria) resulta ser un nodo que se queda estático en la región más alejada dentro de la cobertura del AP más próximo al nodo móvil. Dicha figura fue obtenida con la implementación del algoritmo descrito en el Capítulo 3.

4.1.2. Potencia variable

Ahora bien, si decidimos no elegir la potencia mínima, deberíamos plantear una estrategia diferente. Si definimos una distancia arbitraria con respecto al AP más cercano en la cual queramos simular que nos encontramos, tendríamos que estimar con que potencia debemos transmitir para lograr que la potencia que reciba el AP sea la indicada. Esta potencia depende inversamente de la distancia que definimos. Llamaremos a esta distancia como distancia objetivo.

Comenzemos por definir las siguientes variables:

- La potencia que recibe un nodo móvil desde cierto AP corresponde a $P_{RxAP-nodo}$
- La potencia que recibe un AP de un nodo dado corresponde a $P_{Rxnodo-Ap}$
- La distancia verdadera entre dicho AP y el nodo móvil corresponde a d_{real}

- La potencia nominal corresponde a $P_{TxDefault}$
- La potencia con la que un nodo móvil transmite corresponde a P_{TxNodo}
- La distancia que queremos simular corresponde a $d_{objetivo}$

Comenzaremos por determinar la d_{real} partiendo de la ecuación (4.2):

$$d_{real} = \sqrt[4,0]{\frac{P_{TxDefault}}{P_{RxAP-nodo}}} \tag{4.4}$$

Dado que conocemos dicha distancia podemos definir ahora la potencia que recibirá el AP:

$$P_{Rxnodo-AP} = \frac{P_{TxNodo}}{d_{real}^4} \tag{4.5}$$

El AP llevaria a cabo la estimación de la trayectoria de la siguiente manera:

$$d_{track} = \sqrt[4,0]{\frac{P_{TxDefault}}{P_{Rxnodo-AP}}} \tag{4.6}$$

si sustituimos a la $P_{Rxnodo-AP}$ en esta última ecuación obtendremos lo siguiente:

$$d_{track} = \sqrt[4,0]{\frac{P_{TxDefault} * d_{real}^4}{P_{TxNodo}}} \tag{4.7}$$

, donde:

$$d_{track} = d_{real} * \sqrt[4,0]{\frac{P_{TxDefault}}{P_{TxNodo}}} \tag{4.8}$$

, si hacemos algunos despejes para conocer la potencia de transmisión, obtenemos:

$$P_{TxNodo} = P_{TxDefault} * (\frac{d_{real}}{d_{track}})^4 \tag{4.9}$$

De la ecuación (4.9) obtenemos la potencia de transmisión necesaria para producir una distancia de tracking igual a d_{track}. Si podemos determinar la d_{real} en cualquier momento de nuestra trayectoria, entonces podemos definir una d_{track} cualquiera y establecer un engaño más elaborado como resultado. Un nodo móvil es capaz de simular movimientos distintos e inclusive hasta una trayectoria completamente distinta a la verdadera.

4.2. Nodo virtual

Comenzaremos por definir el concepto de nodo virtual. En la presente tesis consideramos a un nodo virtual como un nodo que existe sólo para la red de comunicaciones y el sistema de estimación de la trayectoria, pero que no existe en el mundo real. Por lo que un nodo virtual cuenta con todos los identificadores y parámetros de un nodo real.

Lo que se pretende lograr con el nodo virtual, es que el algoritmo de estimación de la trayectoria crea que está determinando la posición de un nodo real, mientras que en realidad lo que está obteniendo son trayectorias generadas a partir de los movimientos de un nodo virtual.

Como vimos en el capítulo dos, la idea detrás de un nodo virtual es generar trayectorias ficticias, las cuales lejos de comprometer las trayecorias de los nodos reales, permitan generar las oportunidades necesarias para confundir al sistema de estimación de la trayectoria y éste sea incapaz de determinar las trayectorias de nodos reales.

Para poder explicar mejor este concepto, consideremos un escenario donde sólo tenemos un nodo móvil real que se desplaza implementando un nodo virtual. Este escenario se muestra a continuación en la Figura 4.3.

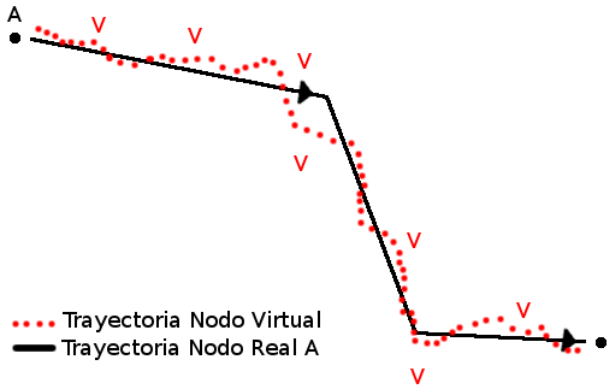


Figura 4.3: Un nodo real con su implementación de nodo virtual.

De la Figura 4.3 es fácil observar que esta implementación de nodo virtual sólo agrega ruido al algoritmo de estimación de la trayectoria que cree ver dos nodos reales. Ahora bien, si consideramos un escenario con dos nodos reales que se cruzan en algún punto de sus trayectoria, podríamos realizar un handover del nodo virtual. Este handover introduciría una variación en el comportamiento del nodo virtual.

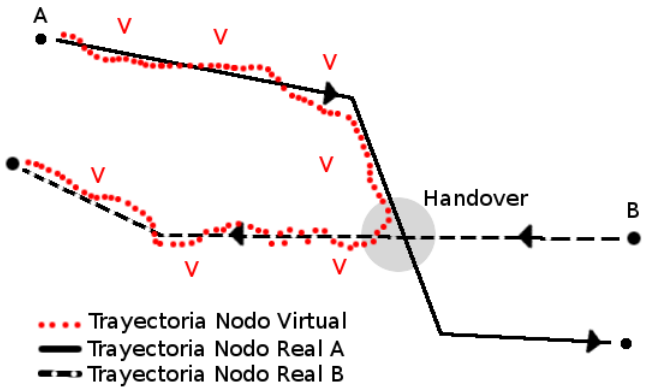


Figura 4.4: Dos nodos reales con un handover de nodo virtual.

En la Figura 4.4 pudimos explotar aún más el concepto del nodo virtual al hacerlo más realista. Sin embargo este comportamiento no brinda protección a la información de la trayectoria de los nodos reales. El verdadero potencial de los nodos virtuales radica en su capacidad para confundir al algoritmo que estima la trayectoria. Para lograr la confusión del algoritmo que estima la trayectoria debemos permitir que un nodo real pueda decidir qué parámetros depositará en el nodo virtual durante el proceso de handover. Por ejemplo, podemos elegir con una probabilidad del 50 % que el nodo real intercambie sus identificadores con el nodo virtual. Este escenario lo podemos observar en la Figura 4.5.

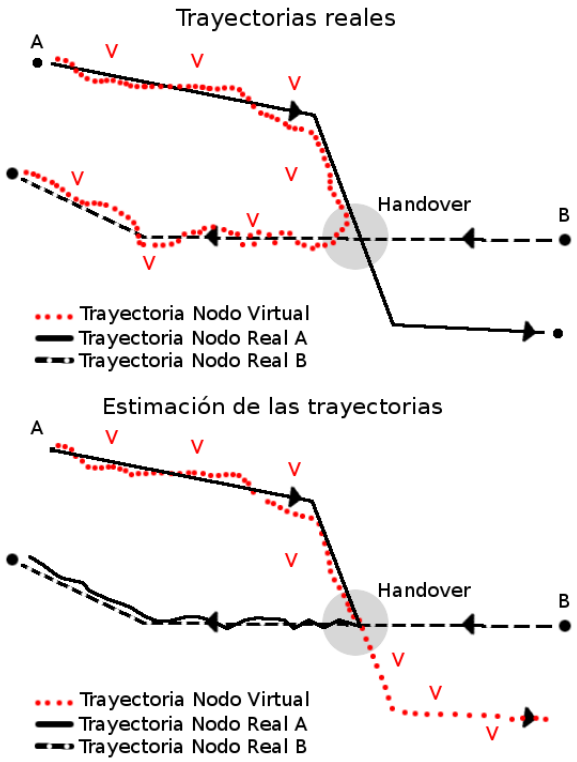


Figura 4.5: Dos nodos reales con un handover de nodo virtual e intercambio de identificadores.

En la Figura 4.5 podemos observar el potencial de la técnia. Esta última consideración para el intercambio de identificadores permite proteger la información de la localización de los nodos móviles reales.

4.3. Criterios para desarrollar los nodos virtuales

Para que un nodo virtual parezca un nodo "real", debe de actuar como tal. El éxito de esta estrategia radica en que el nodo virtual sea tan real como sea posible para el algoritmo de estimación de la trayectoria. Para lograr lo anterior, debemos definir una serie de criterios que aseguren lograr dicho objetivo.

Como vimos en el capítulo dos, una de las principales deficiencias de los nodos virtuales es que quedan descubiertos, o bien es evidente el engaño si se observa al nodo virtual durante un intervalo de tiempo lo suficientemente largo. En nuestro caso, el tiempo de observación se refiere al periodo que el sistema de estimación de la trayectoria recaba muestras del medio inalámbrico para determinar la posiciones geográficas y las trayectorias.

Debemos considerar también que si un nodo virtual es generado a partir de un nodo real.

Este último deberá llevar a cabo todas las funciones que necesite para su operación. Si bien los dos nodos (real y su respectivo nodo virtual) compartiran procesamiento y radio, también compartirían la posición geográfica.

Ahora bien, si partimos de que un nodo real debe generar el engaño por si mismo, esto limita mucho el comportamiento del nodo virtual haciéndolo suceptible a terminar con el engaño de manera prematura. Si en vez de eso, suponemos una interacción coperativa entre los nodos móviles de tal manera que el engaño no dependa de un sólo nodo sino de la colaboración de varios, esto aumentaria las probabilidades de éxito de la estrategia, pero sobre todo minimiza el factor tiempo de observación. Si el nodo virtual continúa en movimiento seguirá generando información equivocada para el sistema de estimación de la trayectoria.

4.4. Creación del nodo virtual

Una vez definidos los criterios a considerar para la creación de nodos virtuales, debemos comenzar a definir la creación de un nodo virtual. Partamos del hecho que el nodo virtual es una copia de un nodo real existente, más específicamente tiene las mismas propiedades que el nodo real que lo generó, es decir potencia de transmisión y ubicación geográfica. Lo único que los hace diferentes es algún tipo de identificador: dirección IP, dirección MAC, etcétera. Por simplicidad sólo diremos que tienen un identificador (id) diferente.

Analizemos el comportamiento de un nodo virtual sólo en términos de presencia, es decir, si la red lo puede ver y cómo lo ve. En esta parte no consideraremos lo concerniente a número de paquetes transmitidos, ni de conexiones, simplemente la presencia y ubicación del nodo virtual.

Un nodo virtual con las características previamente descritas tiene la capacidad suficiente de mantener contacto con la infraestructura de red. Sin embargo el que se encarga de coordinar dicho proceso es el nodo real. Por lo que tenemos que considerar una alternancia entre las funciones del nodo real y las funciones del nodo virtual, para que pueda administrar los procesos de un nodo virtual. Hasta el momento nuestro algoritmo de nodo virtual se parecería más a un nodo real con dos identidades, en vez de un nodo real y su correspondiente nodo virtual.

Surge la pregunta entonces, cómo hacer para que el nodo virtual parezca más un nodo independiente. Mencionamos anteriormente que el nodo comparte la misma potencia y la misma ubicación geográfica, si variamos estas dos características de manera convincente, es posible que logremos crear un nodo virtual creíble, capaz de llevar a cabo el engaño.

4.5. Control de potencia para el nodo virtual

Como explicamos en la sección anterior, cuando queremos manipular en un cierto grado la percepción que el sistema de estimación de la trayectoria tiene sobre un nodo en la red, recurrimos al control de la potencia, ya sea un control mínimo o uno variable. Si dotamos a un nodo real con un control de potencia, esto permitiría aumentar la insertidumbre entre la posición geográfica del nodo virtual con respecto al nodo real que lo generó.

¿Pero cómo se podrían poner a funcionar dichas estrategias juntas? Sabemos que el nodo virtual se encuentra contenido dentro de un nodo real, así mismo, sabemos que el nodo virtual puede tener su propia potencia de transmisión diferente a la del nodo real, es decir que el nodo real tendría que controlar dos potencias de manera independiente y congruente. Mientras el nodo real conserva la potencia de transmisión igual a la máxima indicada por el estándar, el nodo virtual podría implementar un control de potencia mínima o variable.

Si se consigue con éxito tal mezcla, la red que lleve a cabo la estimación de la trayectoria de los nodos detectaría un nodo que se mueve dentro del área de cobertura y uno que no se mueve o que se mueve sobre una ruta diferente.

Si bien un nodo virtual conseguiría una determinada incertidumbre con el control de potencia, este podría no ser suficiente para asegurar por completo una independencia geográfica entre los nodos virtuales y nodos reales. Esto hace necesario introducir el concepto de handover para los nodos virtuales, el cual explicamos a continuación.

4.6. Handover de nodos virtuales

El handover para los nodos virtuales tiene como objetivo introducir cambios creíbles en la posición geográfica de los nodos virtuales con respecto a los cambios de los nodos reales. En esencia, esta estrategia se refiere a permitir que un nodo virtual pueda ser intercambiado entre nodos reales. Esto permitirá que un nodo virtual tenga la posibilidad de reportar coordenadas geográficas muy diferentes a las que podría proporcionar si estuviera asociado a un sólo nodo real.

Para implementar esta estrategia, debemos primero definir algunos aspectos básicos de su operación. Idealmente esta estrategia nos permitirá darle realismo a nuestro nodo virtual, por lo que debemos definir cuidadosamente varios aspectos del handover: ¿cuándo hacerlos?, ¿dónde llevarlo a cabo? y ¿cómo llevarlo a cabo?. Esto determinará que tan convincente es el handover.

Comenzamos por estudiar un mecanismo de control, que nos permita sincronizar a los nodos móviles para poder así llevar a cabo el handover en el momento óptimo.

Al tomar en cuenta a nuestros nodos como elementos reales, notamos un gran número de

limitaciones para llevar a cabo esta sincronía. Un nodo cualquiera no tiene manera de saber que trayectória está siguiendo el mismo, mucho menos saber si continuará en movimiento o se detendrá. Así que debemos valernos de las lecturas que recibamos del medio para llevar a cabo nuestra sincronía.

Nuestra primera aproximación a este problema consiste en tomar en cuenta dos nodos reales que se desplazan dentro de una zona dada, de tal manera que los dos esten al alcance uno del otro. Tomamos entonces las lecturas de potencia recibidas entre los nodos para establecer la proximidad y tratar de predecir el tiempo que continuarán al alcance uno del otro. Esto con el fin de poder establecer una ventana de oportunidad para llevar a cabo el proceso de intercambio del nodo virtual.

Sin embargo, este enfoque presenta varias dificultades. Al medir las diferencias en la potencia que se recibe en uno de los nodos móviles desde el otro nodo móvil podemos conocer la componente radial de la velocidad que está experimentando el segundo nodo con respecto al primero. Para poder completar el modelo sería necesaria más información que nos permita conocer la componente angular de la misma, o bien un punto de referencia conocido.Esta consideración creemos es poco tangible en la realidad ya que un nodo real no puede conocer las coordenadas geográficas de los APs a su alrededor, por ejemplo.

La segunda aproximación que estudiamos, consiste en definir un conjunto de condiciones y sólo si los nodos cumplen dichas condiciones realizar el handover, esto tomando en cuenta que cada condición aseguraría una estabilidad al proceso.

Para establecer un control sobre dichas condiciones definimos una tabla de estados. Esta tabla de estados está formada por un conjunto de variables, las cuales toman valores con respecto a los cambios de potencia que registra un nodo móvil. Los valores serán booleanos, y expresarán el movimiento de aproximación del nodo con respecto a los demás elementos de la red. Como ya dijimos, el conjunto de valores será solamente [1,0]. Asignaremos el valor de uno cuando el nodo se aproxime a otro nodo o AP en la red, y cero cuando se aleje del mismo. Tomaremos en cuenta la diferencia en la potencia de recepción, considerando la potencia actual y la anterior a la misma, esta relación la podemos expresar como:

$$V_{estado,i} = P_{Rx,i} - P_{Rx,i-1} \tag{4.10}$$

$$St(i) = \begin{cases} 1 \text{ si } V_{estado,i} \geq 0; \\ 0 \text{ si } V_{estado,i} < 0; \end{cases}$$

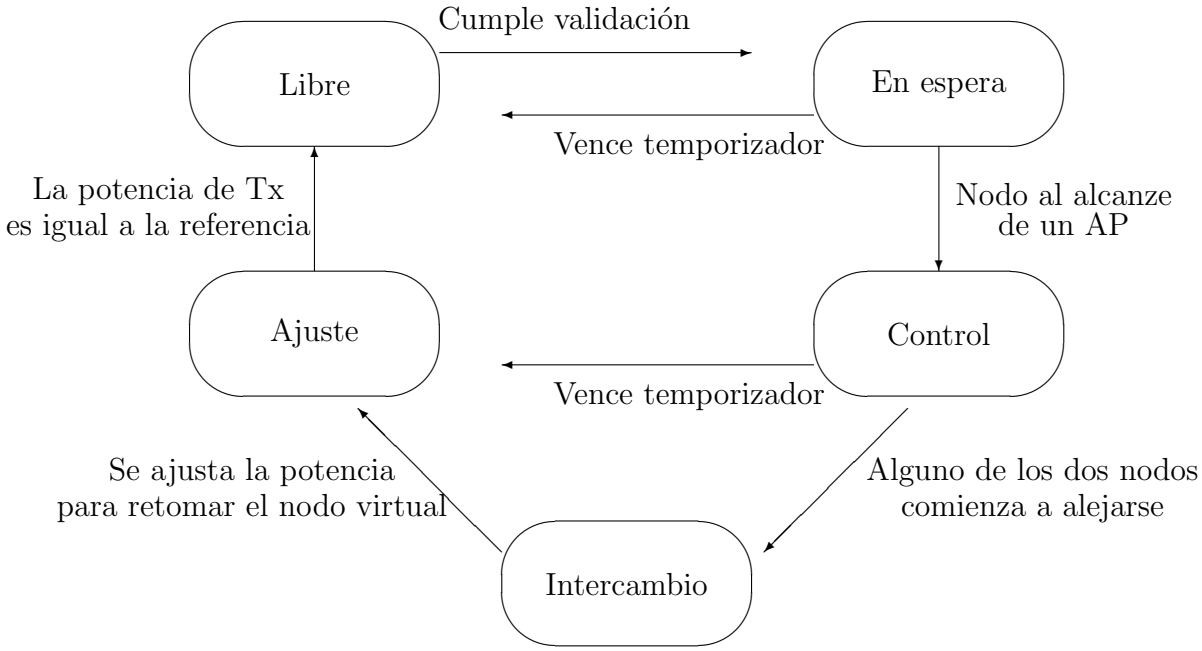
Una vez definidos los estados, podemos construir la tabla de estados. Cada nodo móvil contrendrá una tabla de estados, la cual actualizará conforme se desplaze por el escenario y registre los cambios en la potencia de las señales que recibe. Esta tabla identificará a cada nodo con su ID y su estado correspondiente.

id:1	$st_1 : St(i)$
$\vdots$	$\vdots$
id:n	$st_n : St(i)$

Tabla 4.1: Ejemplo de tabla de estados en un nodo.

Tomando en cuenta la tabla 4.1, definimos un conjuto de estados, que controlen el intercambio de clones entre nuestros nodo móviles. Dichos estados los muestro en el siguiente diagrama.

Diagrama de estados para el control del intercambio del nodo virtual



A continuación describo como funciona el diagrama. Supongamos un nodo a y un nodo b, el nodo a, tiene un nodo virtual y el nodo b, no.

1. El nodo a y b estaran en estado libre hasta que superen la validación. La validación consiste en comparar los ids, en estado 1 (aproximación), de la tabla del nodo a con los del nodo b, si comparten algun id en estado 1, el nodo a pasa al estado en espera.

2. El nodo a permanecerá en espera hasta que se cumpla que sólo tenga dos entradas en su tabla de estados, una correspondiente al id del AP al cual se aproxima y la otra correspondiente a la del nodo b, si no se cumple esto, esperará a que expire el temporizador para estar en este estado y regresa al estado libre.

3. Si el nodo a, cumple con tener sólo esas dos entradas (el AP y el otro nodo b), entonces pasa al estado de control. En dicho estado realizará un control de potencia variable, tomando como potencia objetivo la primer potencia que registro estando en el estado de control, y mandtendrá dicho control hasta que cualquiera de los dos nodos (a o b) cambien el estado del AP de 1 a 0, entonces pasamos al estado intercambio (handover), si no se cumple ninguna de estas condiciones expira el reloj para estar en este estado y se pasa al estado de ajuste.

4. En el estado intercambio se cambian todos los parámetros propios al nodo virtual del nodo a al nodo b y se pasa al estado ajuste.

5. En el estado ajuste se mantendrá el control de potencia hasta que la distancia real, es decir la que existe entre el nodo b y el AP y la distancia objetivo en el control de potencia, sean muy parecidas. Entonces el nodo b deja de ejecutar el control de potencia variable y vuelve a usar la potencia que estuviera usando antes del proceso (la potencia nominal, o bien la indicada por el algoritmo de control de potencia que se estuviera empleando antes del cambio de estado control).

4.7. Resumen del capítulo

En este capítulo explicamos y detallamos las dos estrategias de ofuscación que implementamos en esta tesis. El control de potencia es una estrategia que hace uso de la variación del parámetro de potencia, para poder introducir errores en la determinación de la posición geográfica, siempre y cuando dicha determinación se lleve a cabo por un algoritmo que haga uso de un modelo de propagación. Dentro de esta técnica presentamos dos variantes, el control de potencia mínimo y el control de potencia variable. La primer técnica sólo ajusta la potencia de transmisión a la mínima necesaria para alcanzar al AP más próximo, mientras que la segunda técnica, define una distancia objetivo y ajusta la potencia de transmisión de tal manera que el algoritmo de estimación de la trayectoria, nos ubique en dicha posición.

La segunda técnica de ofuscación que presentamos es la creación de nodos virtuales, dicha estrategia consiste en crear nodos ficticios en la red, de tal manera que su existencia introduzca confusión al algoritmo de rastreo. Esta confusión viene de que los nodos intercambian parámetros, no sólo con los nodos vecinos sino que nodos ficticios también.

Una vez explicadas las técnicas de ofuscación, procederemos a implementarlas dentro de nuestro simulador para cuantificar sus efectos sobre el algoritmo de estimación de la trayectoria elegido para este fin.

Capítulo 5

Implementación

Se eligió al software MatLab para llevar a cabo la implementación de las funciones y características previamente descritas, dado su flexibilidad así como por su sencillez. Corresponde al presente capítulo mostrar la implementación que realizamos del simulador así como de las técnicas de ofuscación.

Para lograr una implementación creíble fueron necesarias ciertas consideraciones, primero debemos de implementar las funciones de la manera más realista posible.

El algoritmo simplificado que elegimos para su implementación es el siguiente:

1. El nodo móvil se desplaza dentro del área de la simulación.
2. El nodo móvil escucha las transmisiones de todos los nodos a su alrededor.
3. El nodo móvil entonces realiza una transmisión.
4. Los APs que escuchan dicha transmición registran dicho evento en un base de datos que contenga id, tiempo y la lectura de potencia que registraron.

Como podemos observar, el algoritmo previamente descrito es sumamennte sencillo y fácil de implementar. También se implementó el algoritmo que describí en el capítulo tres, que corresponde al algoritmo de estimación de la trayectoria. Dicho algorítmo se encuentra anexo en la apendice A donde se muestra su código fuente.

Una vez conseguido lo anterior, se comenzó a implementar el control de potencia mínima. Para llevar a cabo el control de potencia mínima se tenía que modificar la actividad normal del nodo móvil, el algoritmo modificado se muestra a continuación:

1. El nodo móvil se desplaza dentro del área de la simulación.
2. El nodo móvil escucha a todos los nodos a su alrededor.
3. El nodo móvil determina con base en la potencia que recibe de los APs la potencia con la que debe transmitir.

4. El nodo móvil entonces realiza una transmisión con la potencia determinada en el paso anterior.
5. Los APs que escuchan dicha transmición registran dicho evento en un base de datos que contenga id, tiempo y la lectura de potencia que registraron.

Una vez concluido dicho algoritmo, fue necesaria la implementación de un nodo real que llevara a un nodo virtual el cual realizará algún tipo de control de potencia, en este caso se eligió la potencia mínima:

1. El nodo móvil se desplaza dentro del área de la simulación
2. El nodo móvil escucha a todos los nodos a su alrededor
3. El nodo móvil determina si transmitirá con su id o el id del nodo virtual
4. El nodo móvil determina con base en las lecturas la potencia y con el id seleccionado, la potencia con la que debe transmitir
5. El nodo móvil entonces realiza una transmisión
6. Los APs que escuchan dicha transmición registran el evento en un base de datos que contenga id, tiempo y la lectura de potencia que registraron.
7. El nodo vuelve a los parámetros originales.

Para hacer más realista nuestro algoritmo incluimos una herramienta propia del simulador Network Simulator (NS2), la cual genera trayectorias en escenarios móviles basándose en el algoritmo random waypoint. Para el cual sólo definimos un punto inicial, un rango en el cual variará la velocidad, el tamaño del escenario y el tiempo en que se llevará a cabo la simulación. El algortimo se encarga de asignar aleatoriamente trayectorias rectas que le permitan a un nodo ir desde el punto inicial hasta donde la simulación termine.

Mostraré a continuación la función principal de la implementación realizada en MatLab correspondiente al simulador

```
disp('inicia programa')
%%%%%%%parametros
NumAP=100;
NumNodos=1;

v=2; %m/s rapidez
inc_t=1; %0.1 s
step=.5;
```

```
t_ctrl=0; %%%%%cuenta los incrementos en el tiempo

s_sim=400;
ctrl_setdest=0;
ctrl_anim=0;
ctrl_track_visible=[1,1,0,1];
%% vecindad NodMov, vecin AP, track, track solo nodo virtual

status_setdest = 1;  %%[0=>'auto',1=>'manual'];
modes_control = 1;  %%['1=>pmin','0=>intervalo'] aademas a = 0

%%%%define los diferentes figuras dependiendo del tiempo
arr_fig=['d' 'o' 'x' 'v' 'p' 's' 'h' '^'];

%%%%%inicializo vectores para trazar cicunferencias
t=0:0.01:2*pi;
x=cos(t);
y=sin(t);
```

Comenzamos por definir todos los valores necesarios para la simulación. Las variables NumAP y NumNodos determinan el número de access points y nodos móviles disponibles respectivamente en el escenario, así mismo v indica la velocidad constante con la que se moverán nuestros nodos móviles por el escenario. s_sim correspone al tiempo que durará nuestra simulación, dicho parámetro será empleado por el algoritmo de rutas para poder construir nuestro escenario. Defino algunas variables empleadas para controlar el flujo de la animación, tales como *inc_t* que determina los incrementos del tiempo entre evento y evento. La variable *ctrl_setdest* condiciona si se llama o no al algoritmo setdest para que genere una nueva trayectoria para los nodos dentro del escenario. La variable *ctrl_track_visible* es un vector que nos permite indicar que partes de la animación queremos visualizar y cuales dejar de visualizar. El arreglo *arr_fig* define un conjunto de figuras dentro de matlab para denotar los conjuntos de puntos que se hicieron en tiempos diferentes durante la simulación. Por ejemplo, a la primera huella de tracking le corresponderia la figura d (de diamante) para representar todos los puntos correspondientes a dicho conjunto y así sucesivamente. Dado que las zonas de cobertura de un AP son circulares, en nuestra animación necesitaremos una representacipón de dicha circunferencia. En este caso empleamos una circunferencia parametrizada, dichos vectores nos permitirán contruir circunferencias de cualquier radio y en cualquier posición.

```
%%%%%%%estructuras
escucha=zeros(3+NumNodos-1,2);  %%%% lo mas escucha 3 aps
ap_ap_prox=zeros(1,6);
geom=struct('x',0,'y',0,'radio',0);
estilo=struct('color','k','linea','','figura','');
clon=struct('ptx',0,'id',0,'mode','null');
```

```
nodo_m=struct('ptx',0,'id',0,'desc','','geom',geom,'estilo',estilo,
'escucha',escucha,'escucha_cach',escucha,'tabla_estado',escucha,
'chg_st',0,'clon_act',0,'clon',clon,'clon_carrier',0,'ptx_cache',0,
'id_cache',0);
nodo_s=struct('ptx',0,'id',0,'desc','','geom',geom,'estilo',estilo,
'ap_ap_prox',ap_ap_prox);

clon_ctrl=struct('status','vacio','id_carrier',0,'t_proc',0,
't_comm',0,'id_AP',0,'r_clon',0);
```

Con el fin de mantener la modularidad y el orden de las variables dentro del programa, definimos un conjunto de estructuras que representarán a nuestros elementos de la red. Definimos una estructura *nodo_m* que corresponde a un nodo móvil, y una denominada *nodo_s* la cual corresponde a los APs. Podemos observar a simple vista que las dos estructuras comparten algunas similitudes, como son: ptx, id, desc, parámetros geométricos y parámetros de estilo entre otros, estos dos últimos corresponden a elementos propios de la animación.

También podemos observar la definición de una estructura para el control de los clones, la cual utilizaremos como ayuda cuando implementemos el algoritmo de handover. Esta estructura permitirá llevar un registro de la información de control en el proceso.

```
%%%%%parametros prop
PtxDefault=100; %[mW]
PTresh=1e-6; %[mW]
%modelo de prop pr=pt/d^4
dReach=pow2disR(PtxDefault,PTresh);
```

También debemos definir los parámetros de potencia, en este caso debemos definir tanto la potencia nominal de tranmisión (PtxDefault) como la potencia mínima necesaria para detectar una señal (PTresh). Con estos dos valores podemos determinar la distancia de alcanze de cada radiobase o de cada nodo (dReach).

```
%%%%%%llenan las estructuras de acuerdo a los parametros
id_counter=1;
[AP,eje,mapa,id_counter]=info_ap(nodo_s,NumAP,PtxDefault,
dReach,id_counter);
id_counter=id_counter+100; %%para diferenciar nodos
[NodMov,id_counter]=info_nodo(nodo_m,NumNodos,PtxDefault,
dReach,id_counter);
```

Una vez definidos estos parámetros, debemos incluir dichos parámetros en las estructuras de información que definimos como nodos, de eso se encargan las funciones *info_ap* e *info_nodo*. Estas dos funciones toman los valores de potencia y los definen dentro de las estructuras

respectivas, así como también definen el id de cada entidad dentro del escenario.

```
%%%%inicio nodo virtual
NodMov(1).clon_carrier=1;
NodMov(1).clon.id=id_counter;
NodMov(1).clon.ptx=PtxDefault;
NodMov(1).clon.mode = modes_control;
NumNodossim = NumNodos +1;

%%%%nodo virtual empieza en 1 y va a 2
clon_ctrl.status='libre';
clon_ctrl.id_carrier=NodMov(1).id;
```

Definimos quien será el nodo que porte un nodo virtual e indicamos manualmente los parámetros necesarios dentro de la estructura correspondiente. En este caso el primer nodo móvil llevará un nodo virtual activo y si existen más nodos en el escenario y cumplen con el proceso de discriminación entonces el nodo 1 les dará el nodo virtual al nodo real que corresponda. También debemos definir dichos parámetros en la estructura de control para que pueda llevar a cabo el proceso de control.

```
[lx,ly,inicio,d_acum]=rutas_setdest(NumNodos,v,s_sim,inc_t,
eje,ctrl_setdest,status_setdest);
```

En esta línea llamamos a la función setdest y procesamos la información que nos proporciona, para poder traducir los puntos y las velocidades en vectores de trayectoria. Cada punto de la trayectoria se encuentra contenido dentro de las matrices Lx y Ly. Dichas matrices contienen todas las coordenadas x e y, respectivamente, así como también los puntos desde los cuales inician sus recorridos. Dichos puntos están definidos en la matriz inicio.

```
%%%%control de archivos
arch_map=crea_archivador(NodMov);  %%%%metaprograma
pause(10);
archivador

ares=fopen('procesa_cuatro/sceneTrace.txt','w');
fprintf(ares,'%f %f %f %f\n',eje);
```

Como mencionamos previamente, cada radio base debe reportar la infomación que registre a una base de datos, en este caso cada nodo definido en el escenario debe contar con un archivo asociado donde se guardará la información que las radiobases registren de ellos. Hago uso de un meta programa que permita vincular dinámicamente a cada nodo con un archivo de texto, y después llamar dichos vínculos para que el programa pueda hacer uso de ellos. La función *crea_archivador* es un meta programa que escribe un fragmento de código dentro del

programa archivador que después anexo al script principal. De esta manera el vínculo entre archivos y nodos móviles es completamente dinámico. Por último, podemos ver el uso de un archivo denominado “sceneTrace.txt ”. Dicho archivo contiene los detalles para reconstruir todo el escenario, las dimensiones del mismo y las coordenadas de cada uno de los APs dentro del mismo.

```
%%%%%%%%etapa gráfica
if visual
    figure()
    hold on
    for nod=1:NumNodos
        plot(lx(nod,:),ly(nod,:),'k--')
    end

    grid
    axis(eje)
    draw(ctrl_track_visible(2),AP,x,y,ares);
end
%%%%%%%%
```

Comenzamos la parte de la animación, dibujando cada AP con su correspondiente zona de cobertura y las trayectorias que seguirán cada uno de los nodos dentro del escenario.

```
%%%inicializa a los nodos en sus posiciones
NodMov=init_nodo(NodMov,inicio);
```

Este pequeño fragmento de código corresponde a la inicalización de los nodos móviles dentro del ambiente de la animación, definiendo cada nodo en su respectiva posición inicial dentro del escenario.

```
%%%vincula handles graficos a variables
%%%etapa grafica

if visual
    han=zeros(NumNodos,1);
    crea_variables(NumNodos);  %%%metaprograma
    variables                  %%%llama meta programa
    graf_vars
    for nod=1:NumNodos
        set(han(nod),'XDataSource',sprintf('x_vec_%d',nod));
        set(han(nod),'YDataSource',sprintf('y_vec_%d',nod));
    end
    %%%imprimimos la circunferencia desplazada al inicio;
```

```
    if ctrl_track_visible(1)
        ht=zeros(NumNodos,1);
        vecin_nod %%%%define vencindad del nodo 'alcanze'
        graf_vecin
        for nod=1:NumNodos
            set(ht(nod),'XDataSource',sprintf('x_circ_%d',nod));
            set(ht(nod),'YDataSource',sprintf('y_circ_%d',nod));
        end
    end
    hold off
end
```

Igual que en el caso de los archivos vinculados a cada nodo móvil para hacer óptima la animación, es necesario vincular handlers gráficos a un conjunto de variables que indique la posición instantánea de cada nodo mientras esta cambia a lo largo de la animación. Es por esto que recurrimos a otro metra programa que nos permita mantener el código perfectamente dinámico y pueda admitir la simulación de cualquier número de nodos.

```
disp('incia animacion')
tic
[ren_x,col_x]=size(lx);

%%%%%ventanas de texto sobre la animacion
if visual
    h_txneg=text(0,0,'inicio');
    h_txroj=text(0,0,'\color{red}inicio');
end
%%%%%%%%%

rand('state',sum(100*clock));
a = pot_inf;
b = pot_sup;
```

Este fragmento de código marca el inicio de la animación. Definimos el número de puntos en las trayectorias dentro de la variable col_x. También definimos un par de ventanas de texto que indicarán los estados a través de los cuales atraviesan los nodos en el proceso de realización del handover. Por último, definimos el rango de valores dentro del cual variará la potencia en el control de potencia que realize el nodo virtual.

```
graff_pot = fopen('graff_potencia.m','w'); %%%%beta
fprintf (graff_pot,'mat_pot = [ ');
tray_clon = fopen ('trayecto_clon.m','w'); %%%%beta
fprintf (tray_clon,'mat_clon=[ ');
```

Dado que el movimiento del nodo virtual se encuentra condicionado al escenario y a las trayectorias de los nodos móviles, no conocemos su trayectoria real ni tampoco la potencia que haya elegido para transmitir en caso de que se encuentre escogiéndola al azar dentro de un rango determinado. Implementé un par de meta programas más que llevarán el registro de dichos movimientos (potencia y posición) en tiempo real durante la simulación. En este caso definimos un par de archivos donde incluiremos el meta programa con un vector al cual se añadirán elementos mientras vayan ocurriendo los cambios dentro de la simulación.

```
for i=1:col_x
    for nod=1:NumNodos
        %%%actualiza el estado del nodo
        %%%%actualiza coordenadas
        NodMov(nod).geom.x=lx(nod,i);
        NodMov(nod).geom.y=ly(nod,i);
        %%%%actualiza figura
        NodMov(nod).estilo.figura=arr_fig(mod(i,length(arr_fig))+1);

        if visual
            if nod==1
                set(h_txneg,'Position',[lx(nod,i),ly(nod,i)]);
            else
                set(h_txroj,'Position',[lx(nod,i),ly(nod,i)]);
            end
        end

        %%%%actualiza el mapa extendido
        mapa_exten(NumAP+nod,:)=[NodMov(nod).geom.x,
        NodMov(nod).geom.y,NodMov(nod).id];
    end
```

En este fragmento de código inicia propiamente la simulación y la animación. La simulación corresponde a un ciclo *for* que "barre" cada uno de los puntos dentro de la trayectoria de los nodos móviles. Dicho número de puntos está definido en la variable *col_x*. Entonces para cada nodo en cada uno de los ciclos del for debemos actulizar su posición geográfica, la figura con la cual se representarán sus puntos en el algoritmo de estimación de la trayectoria y la posición de las ventanas de texto que deberán moverse junto con cada nodo móvil. Por último, incluimos las posiciones nuevas de cada nodo dentro de un mapa de coordenadas geométricas. Dicho mapa lo utiliza el programa para hacer la búsqueda de los nodos con una mayor rapidez, permitiendo así la fluidez de la animación.

```
    for nod=1:NumNodos
        NodMov(nod)=aire(nod,NodMov,mapa_exten,dReach);
        NodMov(nod)=estado(NodMov(nod));
```

```
        %%%%%variar potencia
    if NodMov(nod).clon_carrier
        if pot_inf==0
            dis_aps = NodMov(nod).escucha(:,2);
            dis_aps = dis_aps (dis_aps>0);
            if isempty(dis_aps)
                a = pot_inf;
            else

                a = (PTresh)*(min(dis_aps))^(4);

            end
        end

        if NodMov(nod).clon.mode
            %%%
            p_temp = a;
        else
            p_temp = a + (b-a).*rand();
        end


        NodMov(nod).clon.ptx = p_temp;
        fprintf(graff_pot,'%f, %f, %f \n',a,p_temp,b);
            %%%meta programa de potencia
    end

end
```

La función "aire" que podemos observar en el fragmento de código anterior, emula las funciones de un nodo que escucha al medio inalámbrico. Dicha función le dice a cada nodo qué nodos vecinos, ya sea AP o nodo móvil, puede escuchar y con que potencia lo hace. Esta función aire guarda dicha información dentro de cada nodo para etapas posteriores de la simulación.

Así mismo la función "estado", define los estados en la trayectoria de todos los nodos vecinos. Como explicamos en el capítulo 4, la estrategia handover necesita de una serie de estados para poder determinar si realizará o no el handover. Dicho estado corresponde a 1 si se aproximan los nodos o 0 si se alejan. Dicha función la realiza la función estado la cual toma en cuenta la tabla de estado anterior y la actual, y define que nodo, ya sea AP o móvil, se aproximó o se alejó del nodo que esta llamándola.

Por último, si el nodo tiene un nodo virtual activo, debemos determinar qué potencia de

transmisión deberá utilizar el nodo virtual. La potencia la definimos dentro del rango $[a, b]$ o bien si $a = 0$ y se encuentra activa la bandera de modo en el nodo virtual, usamos la potencia mínima necesaria para alcanzar al AP más próximo. Sin importar que potencia se haya elegido, dicha potencia es registrada en el meta programa de potencia instantánea.

```
if NumNodos>1
    [clon_ctrl,NodMov]=controlador(clon_ctrl,NodMov,
    pot_inf,pot_sup);
end
%%%%switch clon
NodMov=sw_clon(NodMov,i);

for nod=1:NumNodos
    archivo=obten_arch(NodMov(nod).id,arch_map,
    archivos,NumNodossim);
    reg=transmite(NodMov(nod),NumAP);
    registro(reg,nod,archivo,t_ctrl);

    %%%%%%%%%etapa gráfcia

    if visual
        if ctrl_track_visible(3) ||
            ((NodMov(nod).id==id_counter) &&
                    (ctrl_track_visible(4)))
            %muestra puntos sintetizados
            sintetiza_live(reg,NodMov(nod),AP,step);
        end
    end

    %%%%%%%%%%

     %%%%%%%%%%%%%%%%%%%%
    if NodMov(nod).clon_carrier
        fprintf(tray_clon,'%f %f\n',
            NodMov(nod).geom.x,NodMov(nod).geom.y);  %%meta
    end
    %%%%%%%%%%%%%

    %%%%%%%%%%parte de la etapa visual

    if visual
        status='';
```

```
        if nod==1
            %set(h_txneg,'String',
            sprintf('id: %d \n %d %d',NodMov(nod).id,
            NodMov(nod).escucha(1,:)));
            if NodMov(1).clon_carrier==1,
                    status=clon_ctrl.status; end
            set(h_txneg,'String',
                    strcat(status,mensaje(NodMov(nod))));
        else
            if NodMov(2).clon_carrier==1,
                    status=clon_ctrl.status; end
            set(h_txroj,'String',
                    strcat(status,mensaje(NodMov(nod))),
                            'color','red');
        end
    end

    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end

%%%%%switch clon
NodMov=sw\_clon(NodMov,i);
```

La función "controlador" es la que contiene toda la funcionalidad que definimos en el diagrama de estados. Dicha función evalúa las condiciones de cada nodo y define en que estado se encuentra o bien si debe de cambiar de estado, y por último lo registra en la estructura de control de los clones.

La función "sw_clon" toma en cuenta el número de iteración y el estado del nodo virtual para determinar si activa o no al nodo virtual. Para este programa el nodo virtual solo se activa en números de iteración pares y lo activa antes de llevar a cabo las transmisiónes, desactivandolo después de realizar dicho proceso. En este fragmento del código en especial me refiero a activar o desactivar al pasar los parámetros e identificadores de el nodo virtual al nodo real y viceversa. Es esta alternancia de parámetros y funciones crucial para nuestra simulación, ya que es el tipo de tarea que un nodo real debería realizar en el mundo real.

La función "obten_arch" determina, dado el número de id del nodo que lo llame, que archivo de registro le corresponde.

La función "transmite", como su nombre lo indica, lleva a cabo la funciones de transmisión. Dicha función es una emulación que determina que nodo escucharía dicha transmisión dada la potencia con la cual se llevo a cabo y con que magnitud la registraría. Todo eso será registrado en el archivo respectivo designado por la función "registra".

La función "sintetiza_live" nos permite previsualizar durante la animación las huellas de

localización que va dejando el nodo virtual durante su trayecto por el escenario.

Así mismo se registran las coordenadas x e y en las cuales el nodo real llevó a cabo la transmisión con el nodo virtual como identificador, esto con el fin de poder disponer de la trayectoria real del nodo virtual.

Por último se despliega el mensaje de estado dentro de las ventanas de texto y se regresan los identificadores reales del nodo, guardados en el mismo dentro de la memoria, por los del nodo virtual dentro de la estructura correspondiente.

```
        %%%%%%parte de la etapa grafica
    if visual

        variables
        for nod=1:NumNodos
            %%%%%%actualiza circunferencia del nodo
            vecin_nod
            %%%%%%actualiza los handlers
            if ctrl_track_visible(1)
                refreshdata(ht(nod),'caller')
            end
            refreshdata(han(nod),'caller')
            refreshdata(h_txneg,'caller')
            refreshdata(h_txroj,'caller')
        end
        drawnow
    end
    %%%%%%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    t_ctrl=t_ctrl+inc_t;

    %%%%%%control grafico
    if visual

        pause(ctrl_anim)

    end
end
disp('termina_animacion')
toc
```

Por último, se llevan a cabo las tareas necesarias sobre los handlers gráficos de cada nodo móvil, para que la animación continue en movimiento.

```
fprintf(graff_pot,']; '); %% meta
fprintf(tray_clon,']; '); %% meta
%%%%%%%%%% cierro archivos de tracking

fclose(tray_clon);
fclose('all');
```

Termino con los meta programas de registro de potencia y de coordenadas geográficas y cierro todos los archivos utilizados por el programa principal.

```
[st,re]=system('ls trayecto_clon.m');
if st==0

trayecto_clon


[st,re]=system('ls trayecto_clon.m');
if st==0

trayecto_clon

%%%%%% llamando pos procesamiento
cd ('procesa_cuatro')
tic

error = [0,0,0];

for i_nodos = 1:NumNodos
    if visual
        figure();
    end
    error(i_nodos)=tracking(NodMov(i_nodos).id,
        lx(i_nodos,:),ly(i_nodos,:),inc_t,AP,eje,visual_trak);
    title(sprintf('tracking para nodo %i',NodMov(i_nodos).id));
end

if NumNodossim>NumNodos
    figure();
    carrier = 1;
    if ~NodMov(carrier).clon_carrier
        carrier = carrier + 1;
    end
    error(3)=tracking(NodMov(carrier).clon.id,
        mat_clon(:,1),mat_clon(:,2),inc_t,AP,eje,visual_trak);
```

```
    title(sprintf('tracking para el nodo %i [clon]',
        NodMov(carrier).clon.id));
end
toc
cd ..
```

Llamo por cada nodo en la simualción la función "Tracking", la cual llevará a cabo la estimación de la trayectoria con el algoritmo descrito en el capítulo tres.

```
if visual

    graff_potencia
    figure();

    hold on
    plot(mat_pot(:,1),'-ko');
    plot (mat_pot(:,2),'-kx');
    plot (mat_pot(:,3),'k');
    hold off
    axis ([0 size(mat_pot,1) 0 110])
    xlabel ('eventos')
    ylabel('potencia [mW]')
    title ('potencia elegida durante la simulacion')
    legend ('potencia minima','potencia usada',
        'potencia nominal','Location','SouthEast')
end
```

Por último, si así lo indicamos, mostraremos la gráfica de potencia empleada por el nodo móvil (con el id de nodo virtual) para la transmisión. Dicha información esta contenida en el meta programa que la registró en tiempo real.

5.1. Resumen del capítulo

En este capítulo mostramos de manera detallada el código fuente mediante el cual implementamos nuestro simulador. Dicho simulador se enecargará de construir los escenarios y recabar la información necesaria para que podamos obtener algunas conclusiones acerca de los algoritmos de ofuscación que presentamos en el capítulo anterior.

Capítulo 6

Pruebas y resultados

Corresponde al presente capítulo la presentación y al análisis de los resultados obtenidos en nuestras simulaciones.

Antes de comenzar a presentar las pruebas realizadas, conviene definir los dos parámetros que utilizaremos para presentar los resultados, el primero es el porcentaje de puntos encontrados. El porcentaje de puntos encontrados, se refiere a la cantidad de puntos en la trayectoria encontrada por el algoritmo de localización, tomando como referencia (el 100 %) la cantidad de puntos en la trayectoria real. Este parámetro nos facilita saber, qué tantos puntos de la trayectoria pudo encontrar el algoritmo. El segundo es la distancia promedio de error o bien error, el cual definimos como el promedio de todas las distancias medidas entre cada uno de los puntos en la trayectoria encontrada (por el algoritmo de localización) contra su respectivo punto en la trayectoria real. Para poder asociar cada uno de los puntos en ambas trayectorias (la trayectoria determinada por el agoritmo de localización y la real) tomamos como referencia la marca de tiempo asociada a cada punto. Si en una simulación el porcentaje de puntos encontrados es bajo, podemos asumir que el algoritmo de localización falló, dado que tuvo problemas para determinar la mayoría de los puntos en la trayectoria.

Así mismo, convendría definir los colores que utilizaremos en nuestras figuras. Denotaremos tres elementos de nuestros escenarios: la huella dejada por un nodo, la trayectoria real y la trayectoria estimada por el algoritmo de localización. En el siguiente cuadro muestro los colores empleados.

elemento	**Nodo 1**	**Nodo 2**	**Nodo virtual**
huella	negro	verde	rojo
trayectoria real	azul	azul	azul
trayectoria estimada	cyan	amarillo	magenta

Tabla 6.1: Colores empleados en las figuras presentadas en el capítulo 6

6.1. Desempeño del algoritmo de estimación de la trayectoria

Para las primeras pruebas, comenzamos por evaluar solamente el desempeño del algoritmo de localización. Utilizamos una configuración muy sencilla para dicho propósito. Se considera un escenario con 3 APs y un sólo nodo moviéndose por el área de cobertura siguiendo una trayectoria rectilínea.

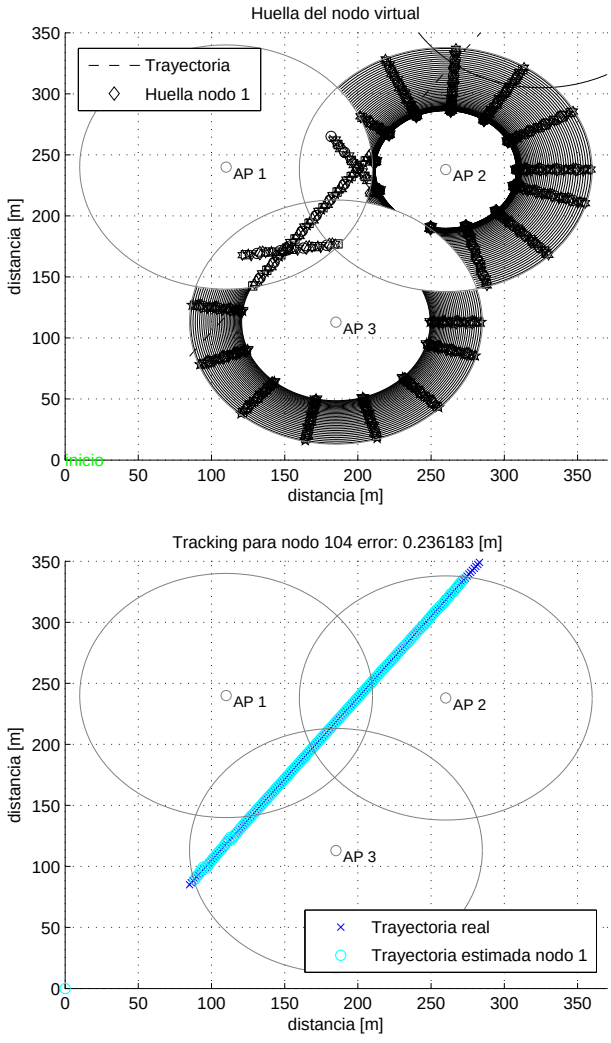


Figura 6.1: Primera prueba de simulación con 3 APs

En la Figura 6.1 podemos observar los resultados de la simulación. En el lado izquierdo podemos ver las huellas dejadas por el nodo móvil durante su trayectoria. Como vimos en los capítulos previos, el algoritmo de localización partirá de las lecturas de potencia y construy-

% de puntos	Error$[m]$
99.68	0.2361

Tabla 6.2: Resultados para la Figura 6.1

irá el mapa que se muestra en esta figura, para finalmente sintetizar la información encontrando así una aproximación a la trayectoria seguida por el nodo móvil. Dicha aproximación se muestra en el lado derecho de la Figura 6.1.

6.2. Desempeño de las técnicas de ofuscación

6.2.1. Control de potencia mínima

Una vez conseguida la simulación exitosa del algoritmo de localización, el siguiente objetivo fue implementar las técnicas de ofuscación planteadas en la presente tesis en el capítulo cuatro. La primer técnica de ofuscación que se definió consiste del control de potencia mínima. Dicha técnica consiste de ajustar la potencia de transmisión de tal manera que el AP más próximo registre siempre la potencia mínima de recepción.

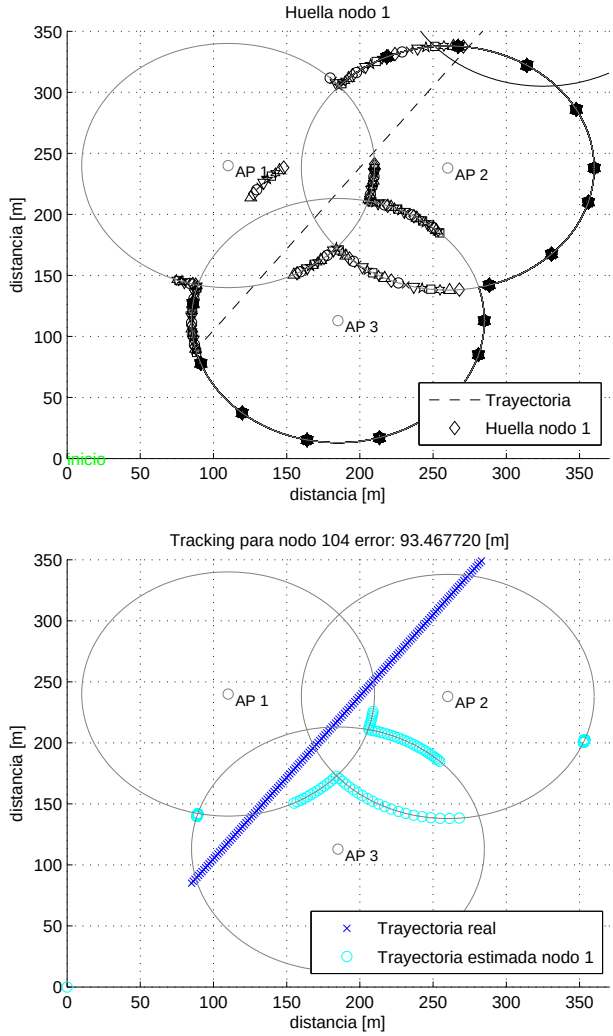


Figura 6.2: Prueba de simulación con 3 APs y un nodo móvil implementando control de potencia mínima

En la Figura 6.2 se pueden observar los resultados de la simulación para un nodo móvil implementando control de potencia mínima. Al igual que la Figura 6.1, en este escenario el nodo móvil sigue una trayectoria rectilínea. Se puede observar en el lado izquierdo de la figura la huella de puntos que dejó el nodo móvil durante su movimiento. A diferencia de la huella dejada en la Figura 6.1, en esta huella no podemos discernir la trayectoria seguida por el nodo móvil. Aplicándole el algoritmo de localizción a este escenario, encontramos los resultados mostrados en la Tabla 6.3

% de puntos	Error$[m]$
98.70	93.4677

Tabla 6.3: Resultados para la Figura 6.2

6.3. Nodo virtual

6.3.1. Nodo virtual con control de potencia mínima

La siguiente prueba consiste en un nodo móvil siguiendo una trayectoria rectilínea, mientras implementa un nodo virtual con control de potencia mínima.

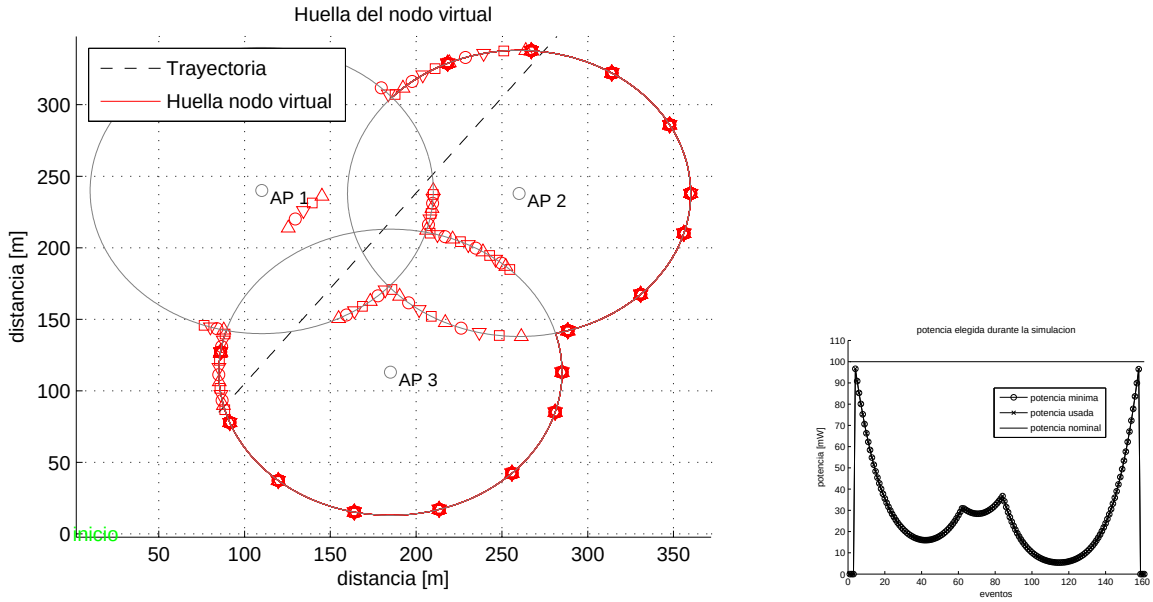


Figura 6.3: Prueba de simulación con 3 APs y un nodo móvil con implementación de nodo virtual

En el lado izquierdo de la Figura 6.3 muestro una captura de la huella dejada por nodo virtual. Podemos apreciar la trayectoria seguida por el nodo móvil real (línea punteada), el escenario, sólo 3 APs en la zona de cobertura y las huellas que dejó el nodo virtual. El nodo virtual implementó un control de potencia mínima. Esto lo podemos corroborar con la imagen del lado derecho de la Figura 6.3, la cual nos muestra la potencia utilizada por el nodo virtual durante la simulación. Además, podemos observar que la potencia usada corresponde con la potencia mínima requerida por el nodo virtual para alcanzar a los APs de este escenario.

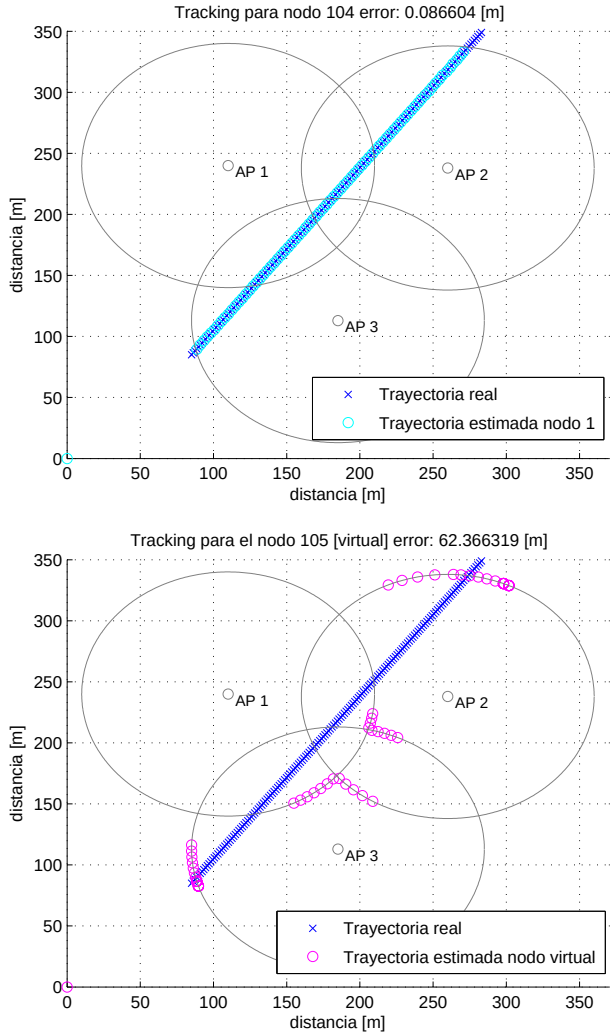


Figura 6.4: Resultados obtenidos por el algoritmo de estimación de la trayectoria en la primera prueba de simulación

En la Figura 6.4 de nuevo tenemos dos imágenes, en la imagen del lado izquierdo muestro el resultado de aplicar el algoritmo para estimar la trayectoria al nodo móvil real. Se grafican simultáneamente la ruta seguida contra la ruta obtenida por el algortimo para estimar la trayectoria. Esta distancia de error corresponde al promedio de todas las distancias obtenidas entre el punto de la trayectoria real contra su correspondiete punto en la trayectoria encontrada por el algoritmo de localización. De igual forma en el lado derecho muestro el resultado encontrado por el algoritmo de estimación de la trayectoria del nodo virtual con control de potencia mínima implementado. Se puede observar una diferencia notable entre los puntos de la trayectoria real contra los puntos de trayectoria encontrados por el algoritmo de estimación de la trayectoria. El resultado del porcentaje de puntos y la distancia de error para la Figura 6.4 se muestra en la Tabla 6.4

lado	% de puntos	Error$[m]$
izq	98.71	0.0866
der	98.70	62.3663

Tabla 6.4: Resultados para la Figura 6.4

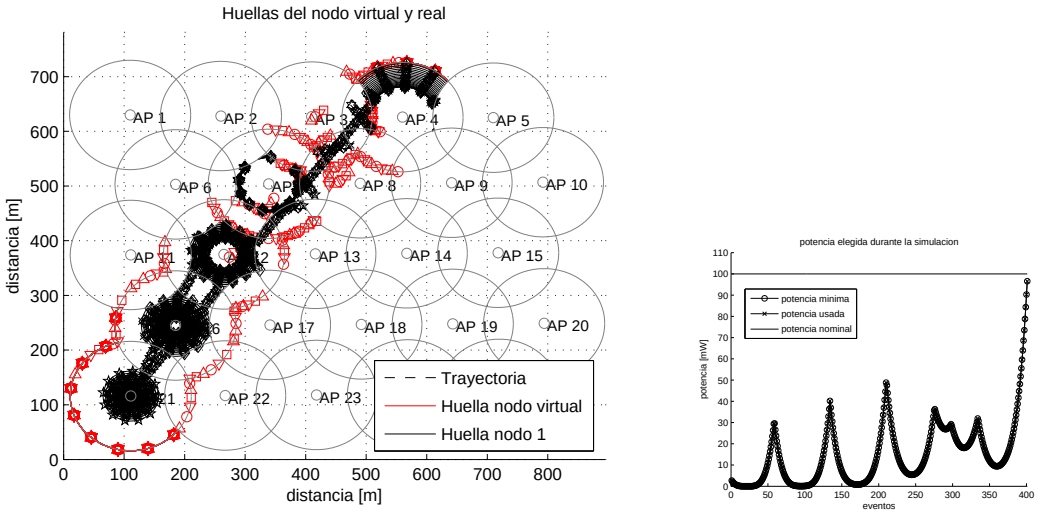


Figura 6.5: Prueba de simulación con 25 APs y un nodo móvil con implementación de nodo virtual

En el escenario mostrado en la Figura 6.5 intentamos averiguar qué es lo que sucede si llevamos a los algoritmos, tanto el de estimación de la trayectoria como el de control de potencia, a un escenario más complicado en cuanto al número de APs. Como podemos observar la trayectoria seguida por los nodos no es complicada, por lo que estos resultados pueden ser comparados con los obtenidos para el escenario de 3APs. En esta figura podemos notar el control de potencia mínima usado también por el nodo virtual.

Decimos que el escenario mostrado en la Figura 6.5 es más complejo que el mostrado en la Figura 6.3. Esta complejidad radica no sólo en el número de APs, sino que el aumento de estos repercute en el número de zonas de alta información para el algoritmo de estimación de la trayectoria, es decir, las zonas donde se intersectan las coberturas de dos o más APs. Como podemos observar en la Figura 6.3 sólo existe una zona de alta información en el centro, mientras que en la Figura 6.5 estas zonas son mayores en número y se encuentran dispersas por todo el escenario. Además, existen APs como los numerados con el id 12, 13 y 14, los cuales tienen zonas de intersección de dos o más APs en todo su perímetro. Esto complica la tarea de los algoritmos de ofuscación dado que reduce las áreas de cobertura de un solo AP en donde tienen una mayor oportunidad de engañar al algoritmo de estimación de la trayectoria.

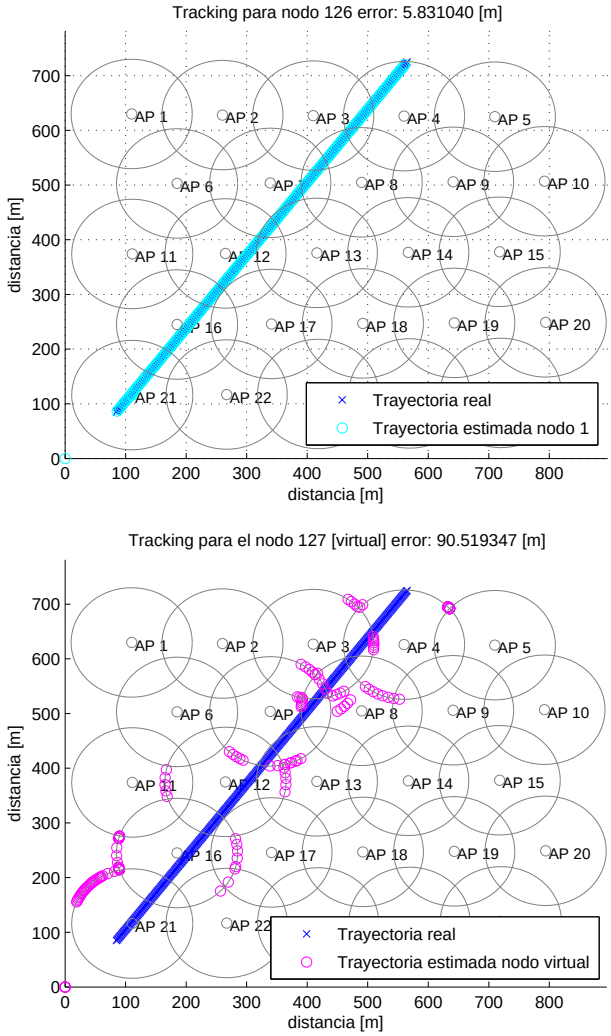


Figura 6.6: Resultados obtenidos por el algoritmo de estimación de la trayectoria para la prueba de 25 APs

En la Figura 6.6 podemos ver, de nueva cuenta, los resultados del algoritmo de localización para ambos nodos. Del lado izquierdo de la Fig 6.6 muestro el resultado de aplicar el algoritmo de estimación de la trayectoria del nodo móvil real, y del derecho el del nodo virtual. Muestro los resultados de la Figura 6.6 en la Tabla 6.5. En este útimo caso, podemos observar una diferencia en cuanto al resultado de distancia de error obtenido para este caso y el mostrado en la Figura 6.4. Esta pequeña variación en el error entre los dos escenarios con 100 APs se debe a que el algoritmo de localización tiende a equivocarse cuando pasa muy cerca del AP. En ese punto la separación, en términos de distancia entre todos los puntos de cada conjunto es menor, y la elección del punto correcto depende de las tolerancias del algoritmo.

lado	% de puntos	Error$[m]$
izq	99.5	5.8310
der	80.59	90.5193

Tabla 6.5: Resultados para la Figura 6.6

6.3.2. Nodo virtual con control de potencia variable

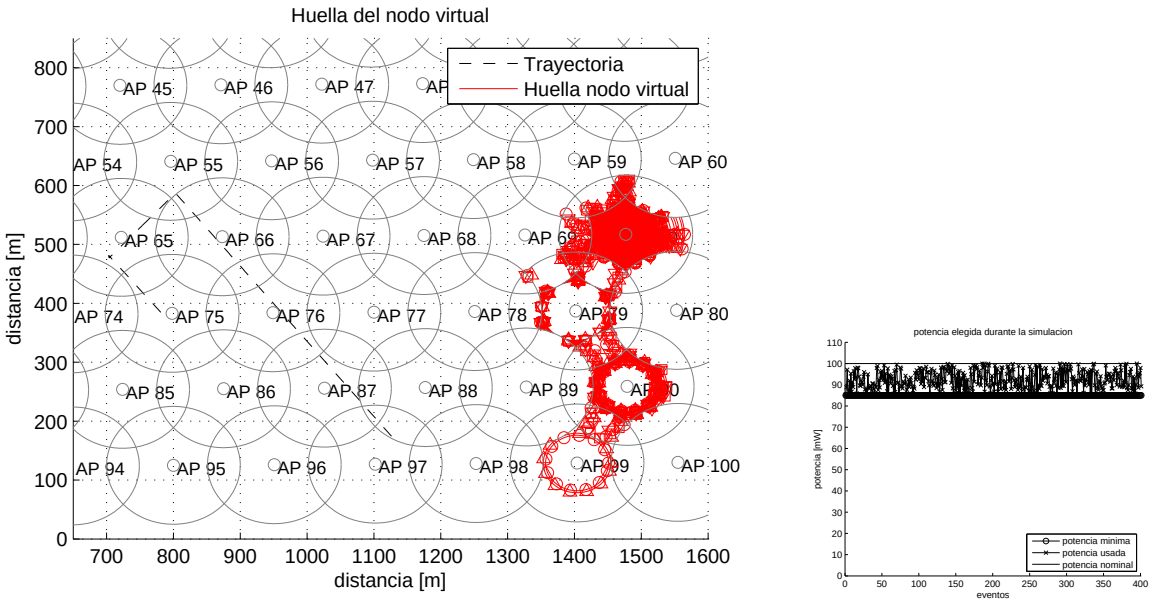


Figura 6.7: Prueba de simulación con 100 APs y un nodo móvil con implementación de nodo virtual

En el lado izquierdo de la Figura 6.7 podemos observar el escenario con 100 APs, sólo que en este caso se simularon dos nodo móviles reales y uno de ellos implementaó un nodo virtual. Del lado derecho de la Figura 6.7 podemos observar la potencia implementada por el nodo virtual. En este caso la potencia varia aleatoriamente en un rango de $[85\,\%, 100\,\%]$. Como podemos observar en el lado derecho de la Figura 6.7, se comprueba que efectivamente el nodo virtual implentó un control de potencia dentro de dicho rango.

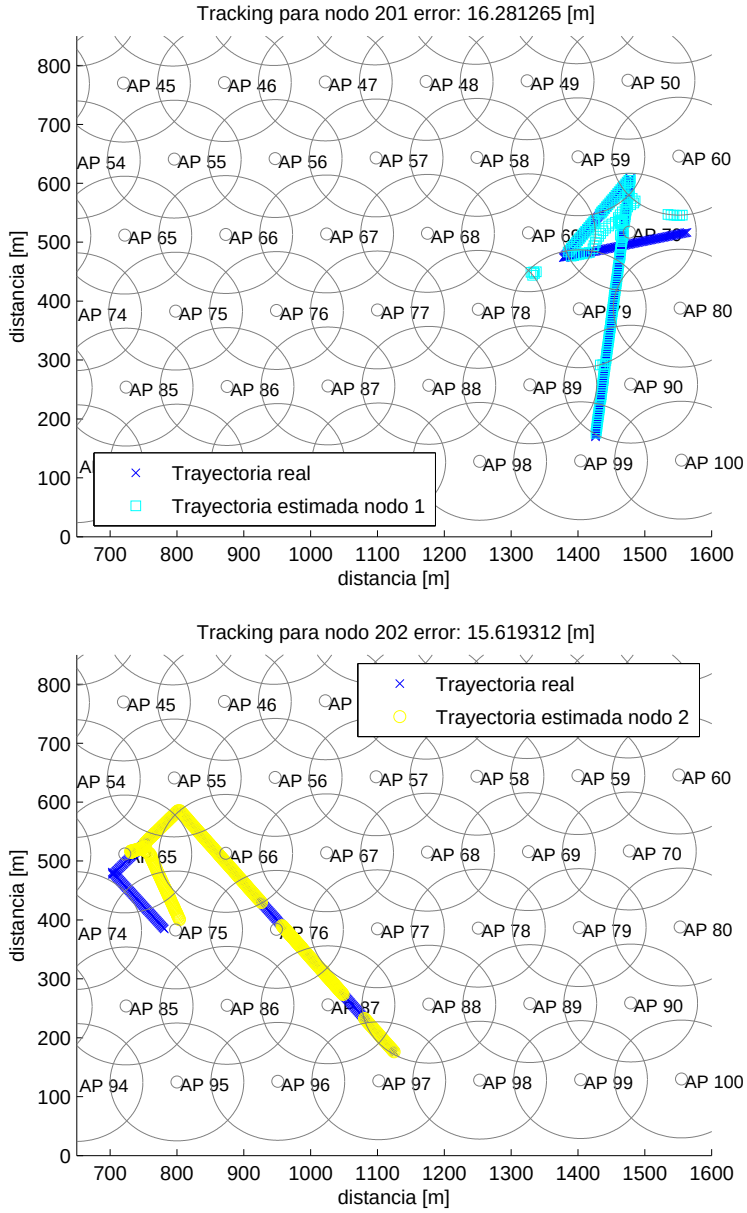


Figura 6.8: Resultados del algoritmo de estimación de la trayectoria con 100 APs y dos nodos móviles desplazándose

En la Figura 6.8 podemos observar las trayectorias encontradas por al algoritmo de estimación de la trayectoria, superpuestas a las trayectorias seguidas por los nodos móviles reales. Los errores son relativamente grandes, pero si observamos la superposición de las trayectorias, el algoritmo de estimación cometió un par de equivocaciones en algunos segmentos de la trayectoria de los dos nodos móviles. Los porcentajes de puntos y la distancias de error para la Figura 6.8 se muestra en la Tabla 6.6.

lado	% **de puntos**	**Error**$[m]$
sup	98.50	16.2813
inf	99.25	15.6193

Tabla 6.6: Resultados para la Figura 6.8

En la Figura 6.8, en la parte inferior podemos ver que el algoritmo de estimó una trayectoria diferente en el último segmento de la trayectoria del nodo 2. Esto porque en dicha región sólo se tiene información de una zona de intersección de dos APs. Como vimos en el capítulo tres esto produce dos líneas rectas con las mismas probabilidades, y sin haber más información, toma como criterio el punto formado por el promedio de todos los puntos de los conjuntos y usa la recta que esté más próxima a dicho punto. Claramente esta no es la decisión adecuada, pero por el momento ese es el único método implementado en el algoritmo de estimación de la trayectoria.

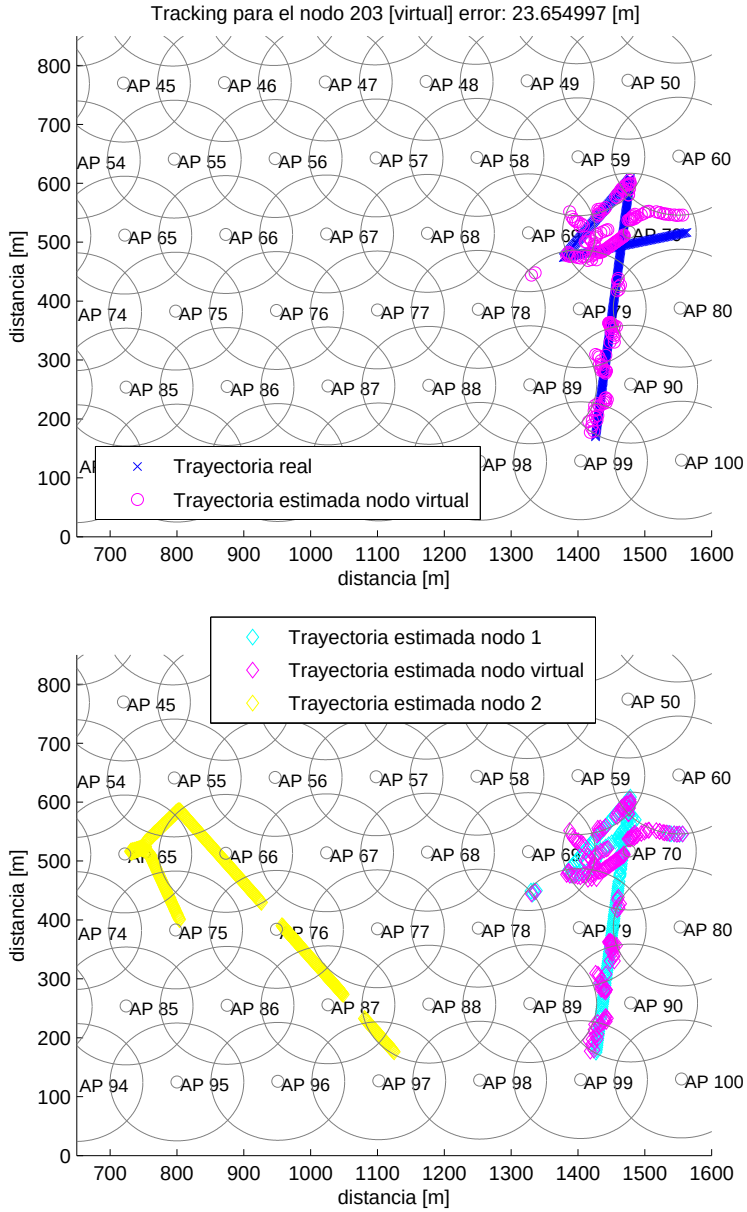


Figura 6.9: Resultados de la estimación de la trayectoria para 100 APs, un nodo virtual y dos nodos reales

En la parte superior de la Figura 6.9 tenemos el resultado del algoritmo de estimación de la trayectoria para el nodo virtual: trayectoria real y la ruta determinada por el algoritmo de estimación de la trayectoria. Así mismo, del lado inferior muestro superpuestas todas la trayectorias obtenidas por el algoritmo de estimación de la trayectoria. Con esta gráfica podemos notar lo que se estaría registrando por el algoritmo de estimación de la trayectoria. Los resultados obtenidos por el algoritmo de estimación de la trayectoria los muestro en la

lado	% de puntos	Error$[m]$
sup	99.50	23.6549

Tabla 6.7: Resultados para la Figura 6.9

Tabla 6.7.

6.3.3. Nodo virtual con handover

Por último, mostraré los resultados obtenidos para las pruebas realizadas al escenario donde se prueba el handover de un nodo virtual. Este escenario resultó ser todo un reto dado que al algoritmo de estimación de la trayectoria se enfrenta a rutas mucho más complejas.

Mostraré primero las figuras Figura 6.10, Figura 6.11 y Figura 6.12. En ellas podemos ver el escenario de simulación y a su lado la potencia que usó el nodo virtual para dejar las respectivas huellas. Podemos notar que para cada uno de los tres casos se empleó un intervalo en la potencia de transmisión que corresponde a uno de tres casos: transmitir al 85 %, transmitir con potencia aleatoria entre $[85\,\%, 100\,\%]$ y por último transmitir con la potencia elegida al azar entre la potencia mínima necesaria y la nominal. El objetivo de analizar el impacto de dicho control de potencia sobre el algoritmo de localización en pro de un nodo virtual más creíble e independiente con respecto a los nodos reales que lo transportan.

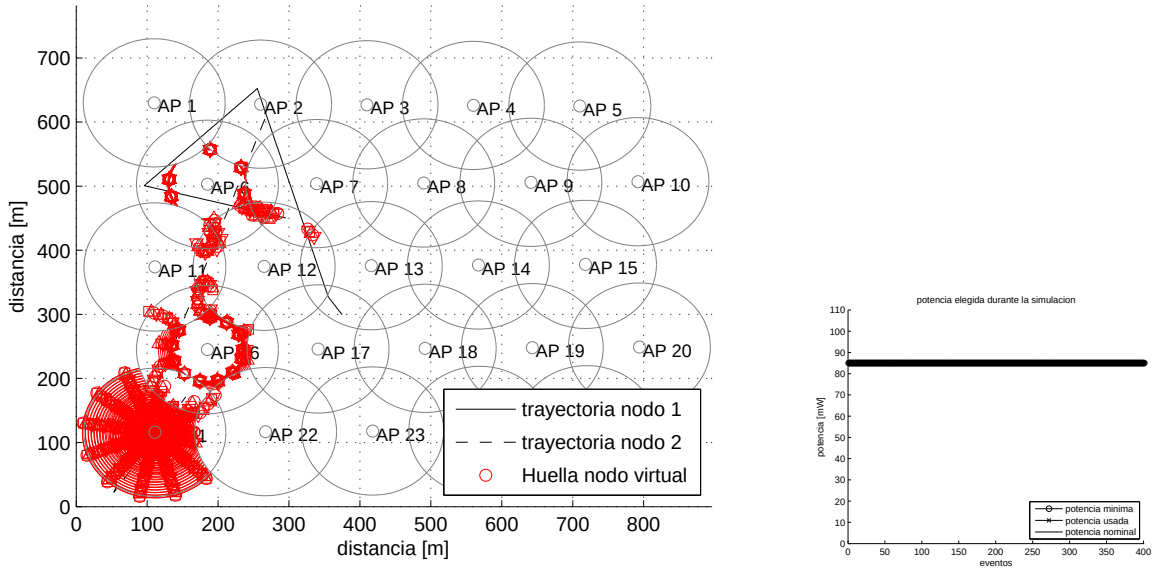


Figura 6.10: Pruebas de handover del nodo virtual para 25 APs con control de potencia 85 %

En la Figura 6.10 muestro el escenario de simulación para la prueba. Se realizó un handover del nodo virtual dentro de un área de cobertura con 25 APs. El nodo realizó un control de potencia fijo en el cual ajustó su potencia de transmisión todo el tiempo al 85 % de su potencia nominal. Se muestra en el lado derecho de la Figura 6.10 la potencia de transmisión empleada todo el tiempo por nuestro nodo virtual.

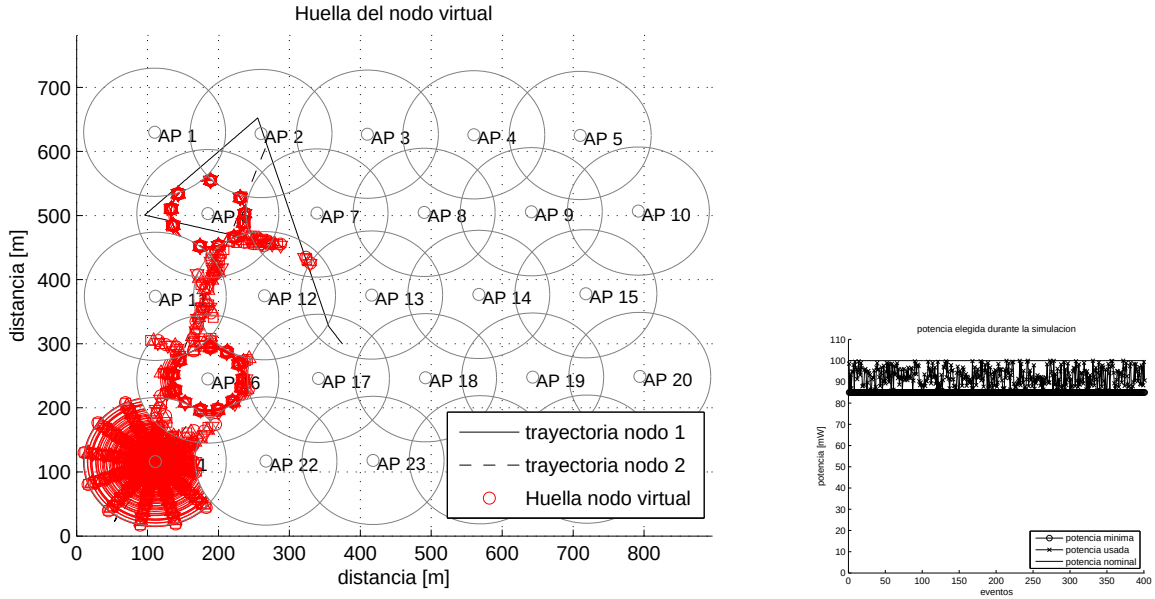


Figura 6.11: Pruebas de handover del nodo virtual para 25 APs con control de potencia [85 %, 100 %]

En el lado izquierdo de la Figura 6.11 podemos observar como variaron levemente las huellas dejadas por el nodo virtual mientras transmitía con la potencia mostrada en el lado derecho de la misma figura, dentro del intevalo [85 %, 100 %]. Los resultados de las distancias de error se mostrarán más adelante.

Sin embargo podemos notar como cambió la ubicación de los puntos de la trayectoria encontrada por el algoritmo de estimación de la trayectoria, el efecto es parecido al obtenido en la Figura 6.10.

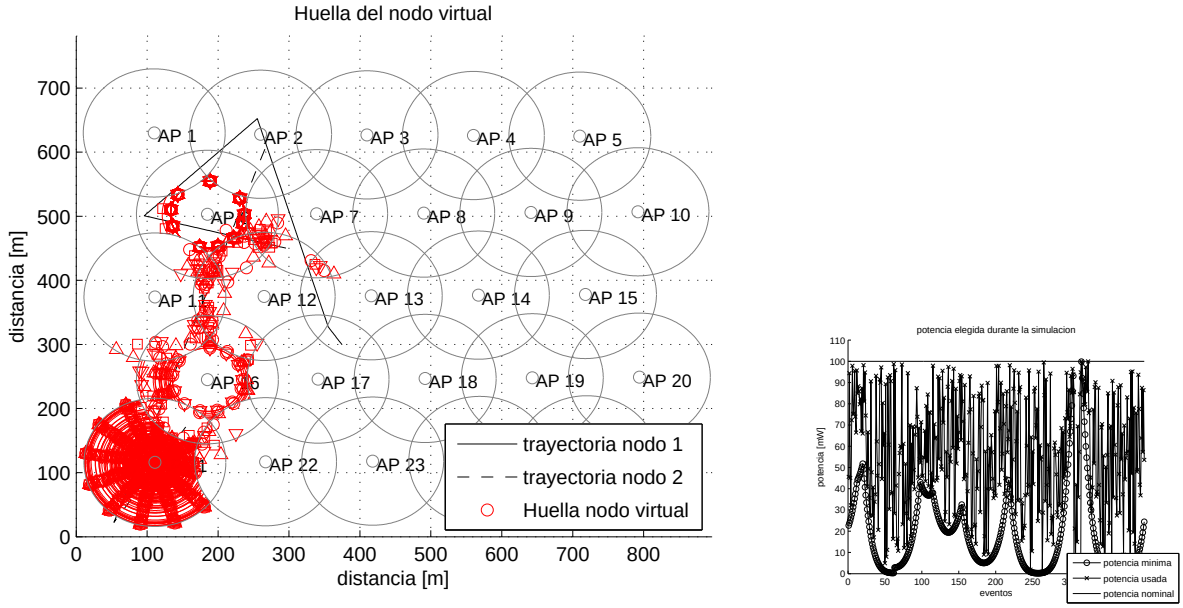


Figura 6.12: Pruebas de handover del nodo virtual para 25 APs con control de potencia $[Pmin, 100\,\%]$

Por último, en la Figura 6.12, vemos el desempeño del algoritmo de estimación de la trayectoria cuando se usa una potencia entre la potencia mínima y la nominal. En este caso es mucho más notorio el cambio en la huella que deja el nodo virtual durante el trayecto.

Este caso mostrado en la Figura 6.12 hace evidente la estrategia del control de potencia, ya que los saltos entre cada uno de los puntos son demasiado grandes. Quizá esto podría ser favorable para la estrategia, pero deja de serlo cuando recordamos que las velocidades de los nodos en este tipo de redes son menores a los dos metros por segundo, en cuyo caso dichas distancias no pueden ser recorridas en los intervalos de tiempo en los que se estimaron.

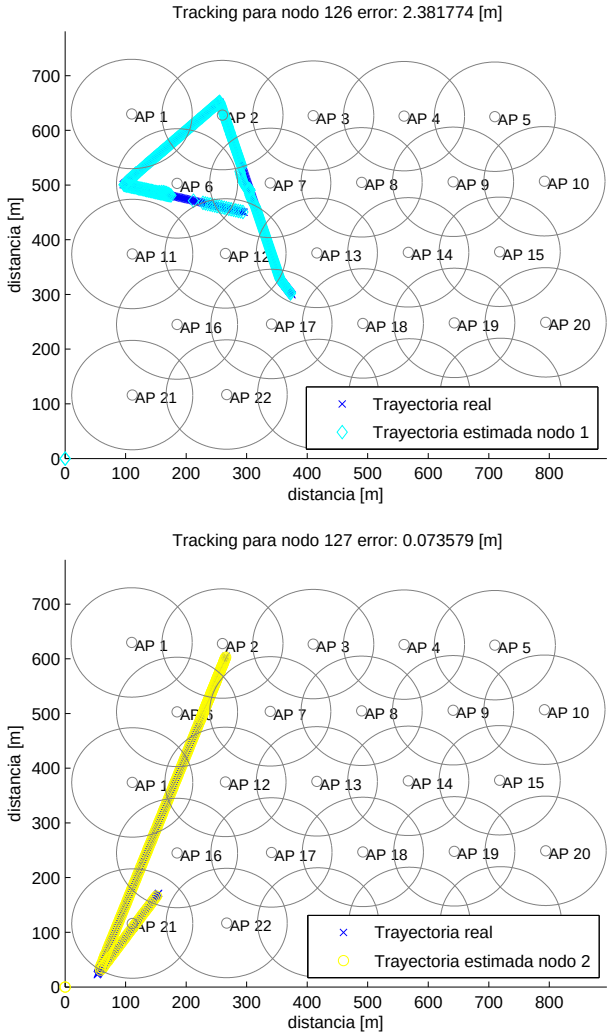


Figura 6.13: Resultados de la estimación de la trayectoria de los nodos 1 y 2 para todos los casos de handover del nodo virtual con 25 APs

Una vez que pudimos observar las huellas dejadas por el nodo virtual en los tres escenarios de handover. Recordemos que en cada escenario el nodo virtual usó una potencia de transmisión diferente. A continuación mostraré los resultados obtenidos por el algoritmo de estimación de la trayectoria para cada uno de los tres escenarios. Sin embargo dado que los nodos reales 1 y 2 realizaron siempre la misma trayectoria, el resultado del algoritmo de estimación de la trayectoria para estos nodos fue siempre el mismo en los tres escenarios, por lo que los mostraremos sólo una vez en la Figura 6.13.

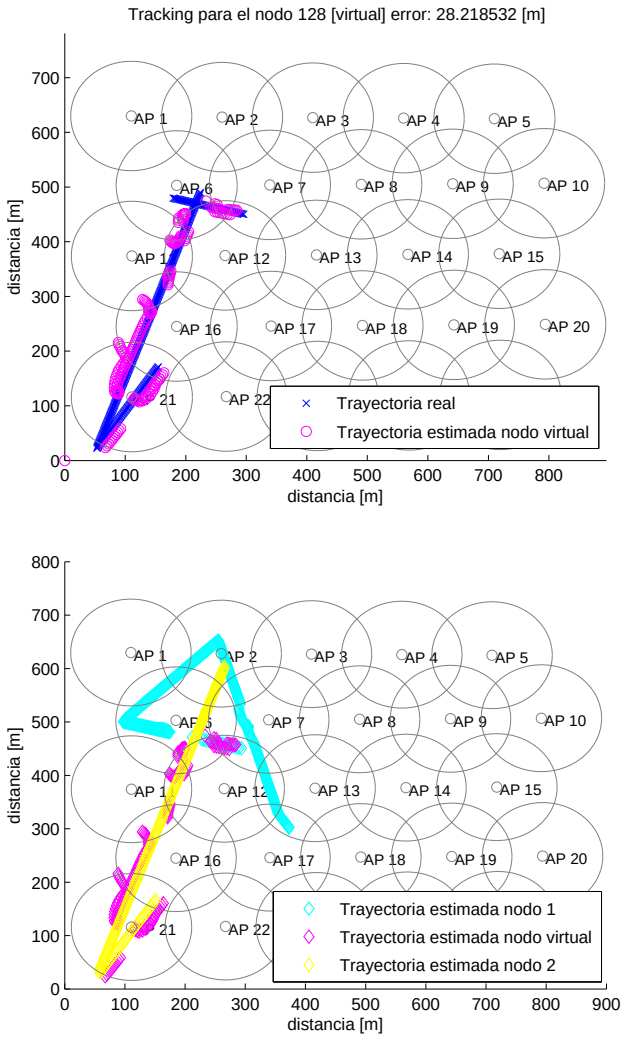


Figura 6.14: Resultados de la estimación de la trayectoria para el handover del nodo virtual con 25 APs y potencia al 85 %

En las figuras: Figura 6.14, Figura 6.15 y Figura 6.16 se muestran los resultados del algoritmo de estimación de la trayectoria para las potencias 85 %, [85 %, 100 %] y $[P_{min}, P_{max}]$ respectivamente. Del lado izquierdo de dichas figuras se muestra la trayectoria real del nodo virtual, superpuesta con la trayectoria encontrada por el algoritmo de estimación de la trayectoria. Mientras que del lado derecho de las mismas figuras, se muestran superpuestas todas las trayectorias encontradas por el algoritmo de estimación, es decir, para cada uno de los nodos móviles y el nodo virtual. Los resultados del porcentaje de puntos así como de distancia de error se muestran en el Cuadro 6.8.

figura	% **de puntos**	**Error**$[m]$
Figura 6.13(izq)	99.45	2.3818
Figura 6.13(der)	99.11	0.0736
Figura 6.14	99.48	28.2185
Figura 6.15	99.48	48.2279
Figura 6.16	99.48	50.1922

Tabla 6.8: Resultados para los experimentos de handover del nodo virutal

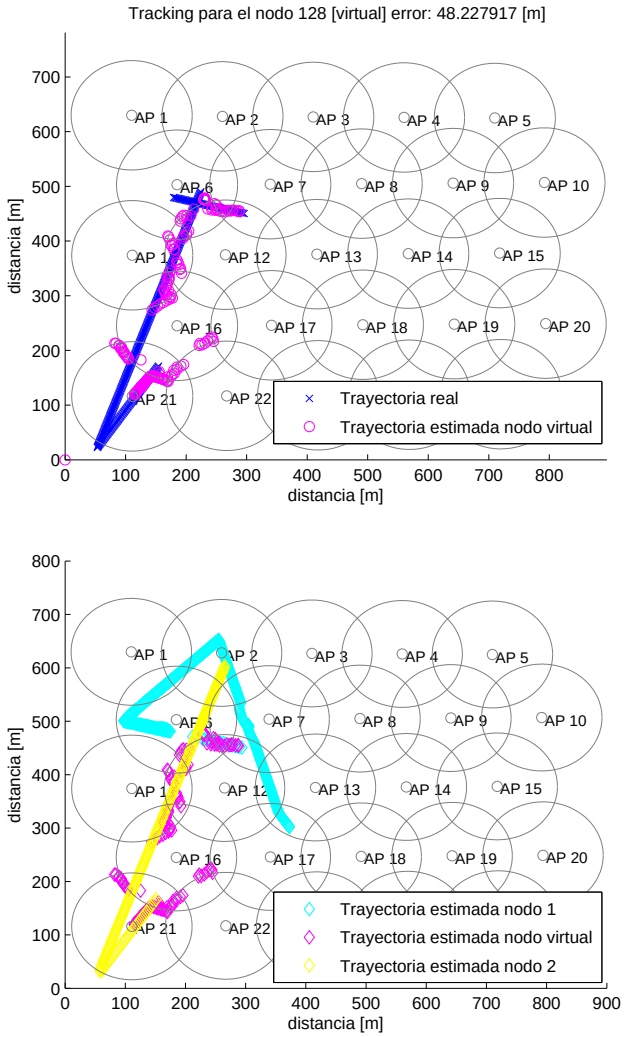


Figura 6.15: Resultados de la estimación de la trayectoria para el handover del nodo virtual con 25 APs en el intervalo $[85\,\%, 100\,\%]$

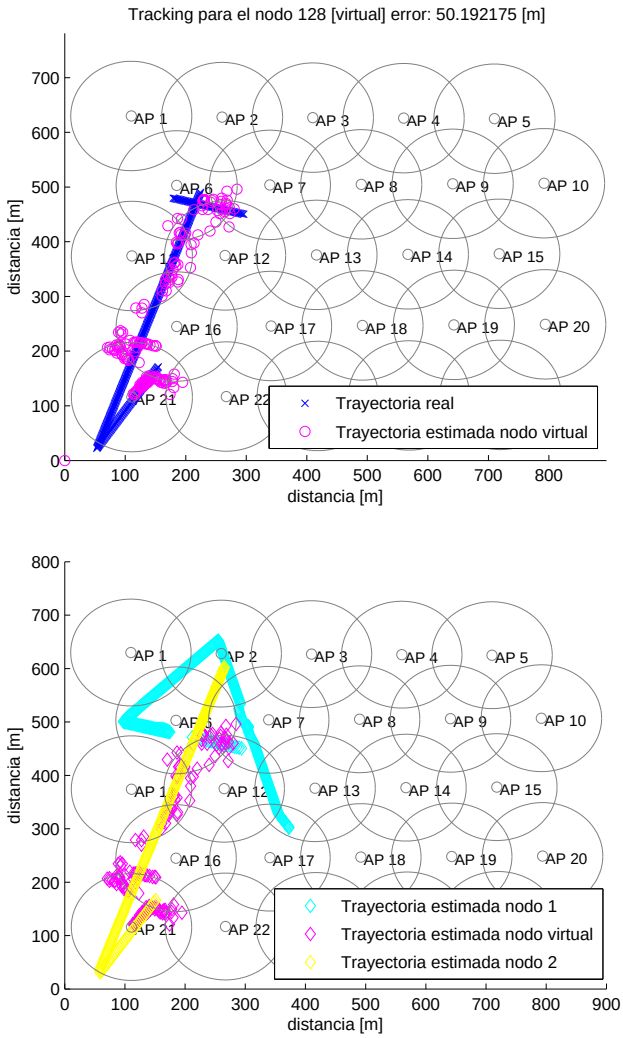


Figura 6.16: Resultados de la estimación de la trayectoria para el handover del nodo virtual con 25 APs en el intervalo $[Pmin, 100\,\%]$

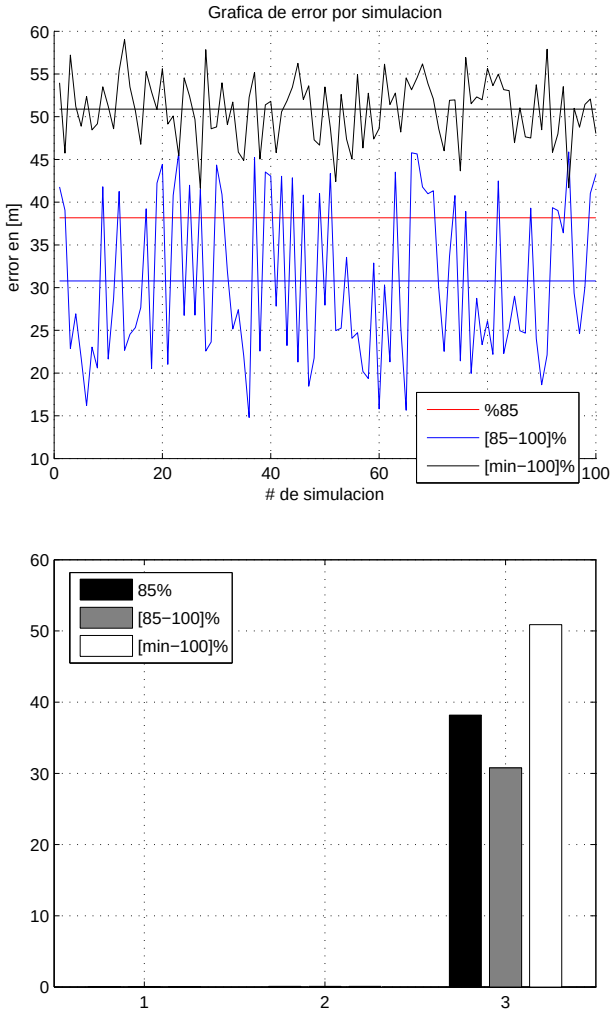


Figura 6.17: Resultados finales de distancia de error para 100 simulaciones con los tres intervalos

Como bien sabemos, al elegir la potencia de transmisión de manera aleatoria en un intervalo dado produce que cada uno de los resultados de la simulación sea único en cuanto a medida de error y trayectoria. La variación de potencia en ciertos puntos críticos de la trayectoria puede afectar el desempeño del algoritmo de estimación de la trayectoria. Este factor puede presentarse o no en los escenarios. Para poder hacer un análisis más completo de estos resultados decidimos llevar a cabo cada una de las pruebas con cada uno de los intervalos considerados previamente cien veces. La gráfica resultante de dicho proceso se muestra en la Figura 6.17. Podemos observar del lado izquierdo de esta figura los resultados de la distancia de error que se presentó en cada una de las pruebas y la gráfica del promedio (en el mismo color). Mientras que en el lado derecho de la Figura 6.17 muestro el promedio de las medidas de error del las pruebas. Cabe resaltar que las pruebas se realizaron sobre el mismo escenario

de handover del nodo virtual que hemos mostrado en el presente capítulo.

Si bien el resultado que presenta la mayor distancia de error promedio es el caso con el intervalo $[P_{min}, P_{max}]$, dicho caso no representa un solución para el objetivo planteado, pues no permite al nodo virtual parecer más independiente con respecto al movimiento de los nodos reales. El promedio de la distancia de error es tan exagerada que parece evidenciar la estrategia del nodo virtual, como podemos ver en la Figura 6.16. Así mismo y observando los resultados de nuestra prueba final, proponemos el uso de los otros dos intervalos de variación de la potencia de transmisión como estrategias para la implementación del nodo virtual. Por último, cabe señalar que dicha estrategia limita el desempeño de los nodos virtuales, dado que reducir la potencia de transmisión siempre tendrá repercusiones en la tasa de transmisión y el alcance del nodo.

6.4. Resumen del capítulo

En este capítulo, presentamos los resultados más sobresalientes que encontramos durante la etapa de pruebas de nuestro simulador, podemos observar las distintas situaciones a las que nos enfrentamos durante el desarrollo de la propuesta de esta tesis. Prensentamos los resultados en orden de complejidad, primero para un escenario de 3AP, 25AP y 100APs con trayectorias simples, después con un escenario de 25 APs con trayectorias complejas y handover del nodo virtual.

Capítulo 7

Conclusiones

Como vimos en el primer capítulo, diferentes personas tienen diferentes necesidades en cuanto a su información de la localización. Es por esto que se han propuesto cuatro esquemas para brindar seguridad a los usuarios de los servicios basados en la localización. En esta tesis nos enfocamos en la ofuscación como medio para lograr brindarle seguridad a los usuarios de las redes de acceso inalámbrico.

En el segundo capítulo pudimos revisar artículos internacionales relevantes para esta tesis que desarrollan tanto técnicas de localización como de antilocalización. En dichos articulos pudimos observar las principales ventajas y desventajas de dichas técnicas así como también, pudimos definir algunos de los criterios para realizar nuestras propuestas de algoritmos de localización y antilocalización.

Por su parte en el tercer capítulo llevamos a cabo el planteamiento y desarrollo del algoritmo que estima las trayectorias de nodos móviles dentro de un escenario inalámbrico. Dicho algoritmo hace uso de las lecturas de potencia que toma de medio inalábrico así como también las posiciones de las radiobases que llevan a cabo las lecturas para poder determinar la localización de los nodos móviles.

En el cuarto capítulo desarrollamos la propuesta de las dos técnicas de ofuscación presentadas en esta tesis. Dichas técnicas de ofuscación son: el control de potencia y la creación de nodos virtuales. El objetivo de dichas técnicas es ofuscar la localización llevada a cabo por el algoritmo que estima la localización.

En el quinto capítulo implementamos las técnicas de ofuscación propuesta en el capítulo cuatro. Nos enfrentamos a una serie de retos para lograr esta tarea, dado que el nivel de realismo del simulador era fundamental para considerar los resultados como satisfactorios. Uno de las principales dificultades fue la de el desarrollo del diagrama de estados finitos que controla el handover de los nodos virtuales. Este diagrama nos permitió solucionar el problema del control del handover, dado que la estimación de la posición entre dos nodos móviles que no cuentan con equipo especializado de localización (ej. GPS) resultó ser muy complicado.

En el quinto capítulo se presentó también el modelo de propagación más utilizado dentro de las WLAN, sin embargo el simulador fue desarrollado bajo el concepto de que cualquier

modelo de simulación que permita relacionar distancias con potencias y viceversa pueda ser utilizado por lo que los alcanzes de las técnicas podrían aplicarse a cualquier escenario inalámbrico no sólo WLAN.

En el sexto capítulo mostramos las pruebas y resultados más siginificativos que desarrollamos durante esta tesis. Pudimos constatar los efectos producidos por las técnicas de ofuscación aquí presentadas. También pudimos evaluar el desempeño tanto del algoritmo de estimación de la trayectoria como de las técnicas de ofuscación. Encontramos que el control de potencia mínimo por sí sólo no mitiga completamente los efectos del algoritmo de estimación de la trayectoria, ya que si bien introduce una incertidumbre de algunas decenas de metros en escenarios con alta densidad de APs aún permite formar una trayectoria. Algunos de los efectos de la técnica de ofuscación control de potencia mínima son que si bien disminuye la potencia de transmisión empleada, disminuye la tasa de transferencia de la información, ya que obliga al nodo a cambiar el esquema de modulación por el más robusto posible.

En cuanto a la técnica de creación de nodos virtuales se comprobó el funcionamiento de la técnica de handover la cual permitiría la confusión del algoritmo de estimación de la trayectoria, lo que trae como consecuencia que el sistema de localización no tenga información real de los nodos en la red.

Se definieron tres estrategias de control de potencia. La primera fue reducir la potencia al 85 % , la segunda fue variar la potencia de manera aleatoria en el intervalo entre 85 % y el 100 % y por último entre el valor instantánemo mínimo y el 100 % elegido al azar. Sin embargo los resultados mostrados capítulo seis indican que el error promedio para dichos intervalos son de: $28m$ para el caso de 85 %, $32m$ para el intervalo de $[85\,\%, 100\,\%]$ y por último de $49m$ para el intervalo de $[Pmin, 100\,\%]$. Este último caso, a pesar de haber dado el error promedio más elevado que los demás, no lo consideramos como una solución viable, dado que los puntos en la trayectoria hacen evidente el empleo de la técnica de ofuscación.

Dentro de los objetivos planteados en esta tesis tenemos la definición y la evaluación de dos técnicas de ofuscación, las cuales se llevaron a cabo durante el desarrollo de la presente tesis. La implementación del algoritmo atacante fue descrita en el capítulo tres y el algoritmo se presenta en la apendice A. La implementación del simulador completo, incluyendo los nodos, se desarrollo en al capítulo 5 y el algoritmo completo se presenta en el capítulo 4. Los resultados del uso de las técnicas de ofuscación se llevaron a cabo en al capítulo 6 de la presente tesis.

Por todo lo expuesto en esta tesis considero que se cumplieron con todos los objetivos planteados al inicio de la misma.

Apéndice A

Código fuente del algoritmo de estimación de la trayectoria

```
%%%

function [ecm,ret]=tracking(id_obj,lx,ly,inc_t,AP,eje,visual)
ecm=0;
disp(sprintf('inicia pos procesamiento de %d...',id_obj));
disp('buscando archivos necesarios...');
[st,msg]=system(sprintf('ls registro_%d.txt',id_obj));
if ~st
    %figure();
    disp('construyendo escenario...')
    %ap=draw_scene('procesa_cuatro/sceneTrace.txt');

    if visual
        hold on
        axis(eje);
        draw_scene_ap(AP);
        plot(lx,ly,'x');
        title('Potencia constante')
        hold off
    end

    %%%extrae potencias desde archivos
    [t,p,id]=define_puntos(sprintf('registro_%d.txt',id_obj));
    t_vec_orig = t;
    %ap=ap(:,[2,3,5]); %%define mapa de aps

    %% cuenta conjuntos

    i=0;
    cont_uno=0;
```

```
cont_dos=0;
cont_tre=0;
while (i<max(t))
    mat=(t==i);
    switch(sum(mat))
        case 1
            %disp(sprintf('1: %d',id(mat)));
            cont_uno=cont_uno+1;
        case 2
            %disp(sprintf('2: %d %d',id(mat)));
            cont_dos=cont_dos+1;
        case 3
            %disp(sprintf('3: %d %d %d',id(mat)));
            cont_tre=cont_tre+1;
    end
    i=i+inc_t;
end

%%%%ya tenemos los tamaños de los vectores
circ=struct('id',0,'t',0,'x',0,'y',0,'r',0);
    %%estructura de circunferencia

conjuntos_uno=repmat(circ,cont_uno,1);
conjuntos_dos=repmat(circ,cont_dos,2);
conjuntos_tre=repmat(circ,cont_tre,3);

%conjuntos_uno=zeros(cont_uno,1);
%conjuntos_dos=zeros(cont_dos,3); %id_uno, id_dos, conjunto
%conjuntos_tre=zeros(cont_tre,3); %id_uno, id_dos, id_tre

%%forman conjuntos

%%%%formar conjuntos
i=0;
f_u=0;
f_d=0;
f_t=0;
while (i<max(t))
    mat=(t==i);
    switch(sum(mat))
        case 1
            f_u=f_u+1;
            val_id=id(mat);
            val_p=p(mat);
```

```
            for j=1:length(val_id)
                conjuntos_uno(f_u,j).id=val_id(j);
                conjuntos_uno(f_u,j).x=AP(val_id(j)).geom.x;
                conjuntos_uno(f_u,j).y=AP(val_id(j)).geom.y;
                conjuntos_uno(f_u,j).r=pow2dis(val_p(j));
                conjuntos_uno(f_u,j).t=i;
            end
        case 2
            f_d=f_d+1;
            val_id=id(mat);
            val_p=p(mat);
            for j=1:length(val_id)
                conjuntos_dos(f_d,j).id=val_id(j);
                conjuntos_dos(f_d,j).x=AP(val_id(j)).geom.x;
                conjuntos_dos(f_d,j).y=AP(val_id(j)).geom.y;
                conjuntos_dos(f_d,j).r=pow2dis(val_p(j));
                conjuntos_dos(f_d,j).t=i;
            end
        case 3
            f_t=f_t+1;
            val_id=id(mat);
            val_p=p(mat);
            for j=1:length(val_id)
                conjuntos_tre(f_t,j).id=val_id(j);
                conjuntos_tre(f_t,j).x=AP(val_id(j)).geom.x;
                conjuntos_tre(f_t,j).y=AP(val_id(j)).geom.y;
                conjuntos_tre(f_t,j).r=pow2dis(val_p(j));
                conjuntos_tre(f_t,j).t=i;
            end
    end
    i=i+inc_t;
end

%%definir situacion
if (cont_dos+cont_tre)>4
    %%funcion normal
    %%%intetizar dos
    %%%intetizar tres
    if cont_uno>2
            %%%sintetitzar uno
    end
    %%encontrar el numero de rectas con conjuto de dos
    %%por cada recta obtener ecuacion
    %%filtrar puntos de uno
```

```
        %% %medir error
    elseif (cont_uno>2)
        %% %se hace un regresion sobre todos los puntos sintetizados
        %% %sintetizar uno
        %% %obtener ecuacion de recta
        %% %filtrar puntos
        %% %medir error
    else
        disp('no hay suficientes puntos para procesar');
    end

%%%%%%%%%%%%%pendiente
%%encapsular funciones para hacer mas sencillo el main tracking




    %%%%%%%%%%%%%%%%%%%%%%%%
    step = 0.01;
    n_con_uno = ceil(2*pi/step);
    puntos_todos = zeros(n_con_uno*cont_uno + 2*cont_dos + cont_tre, 3); % [t
    puntos_dos= zeros (cont_dos,4); %[t,x ,y,linea]
    ind_puntos = 0;
    puntos_final = zeros (length( unique(t)),3);
    puntos_final(:,1)= unique(t);
    %%%%%%%%%%%%%%%%%%%%%%%%

     %% sintetizo para dos

     pts_dos = sinteDos(cont_dos,conjuntos_dos);

     %colores=['r','b','k','y','c','m','g'];
     colores=['r','k','c','m','r','k','g'];
     colorcito = colores(mod(id_obj,7)+1);

     %hold on
     %for jp=1:cont_dos
     %     res=pts_dos(jp,:);
     %     color=colores(mod(res(6),7)+1);
     %     plot(res(2:3),res(4:5),sprintf('x%s',color))
     %end
     %hold off
```

```
 %% division de las rectas por vector unitario
 vec_tot_dos = lineaDos(cont_dos, pts_dos);

  %% determinación de las rectas
 puntos_final=finalRectaPuntosDos(puntos_final, vec_tot_dos);

    %% %sintetizo los puntos y dado que su prob es 100%
    %%pasan directo a los puntos finales
 puntos_final=sinteTres(puntos_final, conjuntos_tre);

 %% filtra los ultimos puntos con base en las rectas entre los

 puntos_de_uno = sinteUno(cont_uno, conjuntos_uno, AP, circ);
 mat_rectas = determinaRectas(puntos_final, t, id, AP, puntos_de_uno, inc_t)
 puntos_final = sinteFinal(puntos_final, mat_rectas, puntos_de_uno, inc_t)


%% impresion final
if visual
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  hold on
  plot (puntos_final(:,2), puntos_final(:,3), sprintf('o%s', colorcito))
  hold off
 %puntos_final
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
end

%       for i=1:length(puntos_final(:,2))
%   hold on
%   plot (puntos_final(i,2), puntos_final(i,3), sprintf('o%s', colorcito))
%pause(1)
%   hold off
%end
 %% medicion de error

   d_error = zeros(size(puntos_final,1),1);
   puntos_ref = zeros(length(lx),3);
   puntos_ref(:,1) = [0:inc_t:length(lx)-1];
   puntos_ref(:,2) = lx';
   puntos_ref(:,3) = ly;


   t_ref = puntos_ref(:,1);
```

```
a= 0;
b =0;
    for i_d =1:size(puntos_final,1)
        ref = puntos_final(i_d,:);
        if ref(2)>0 && ref(3)>0
            %%%
            if sum(t_ref==ref(1))>0
                d_error(i_d)= dista([ref(2),ref(3)],
                                    [puntos_ref(t_ref==ref(1),2),
                                    puntos_ref(t_ref==ref(1),3)]);
                a = a+1;
            else
                disp('no hay tiempos');
            end
        else
            disp('punto perdido');
                b=b+1;
        end
    end
        disp(sprintf ('pts totales:%f puntos perdidos:
%f procentaje:%f',a+b,b,a/(a+b)))
   d_error = d_error(d_error>0);
   ecm = mean(d_error);
else
    disp('error en archivo de registro:')
    disp(msg)
end

ret = puntos_final;
```

Glosario

Active Bat sistema de posicionamiento ultrasónico de baja potencia. 7, 13

AoA ángulo de llegada (Angel fo Arrival). 14

AP puntos de accesso (Access Points). 15

GPS sistema de posicionamiento global (Global Position System). 2, 7, 8, 13

GSM sistema global para las comunicaciones móviles (Global System for Mobile Communications). 13

GUIDE Guiding Users in Distributed Enviroments. 14

IEEE Instituto de Ingenieros Electricistas y Electrónicos (Institute of Electrical and Electronics Engineers). 2

IP Protocolo de Internet (Internet Protocol). 14, 28

MAC Media Access Control. 28

NS2 Network Simulator. 41

OSI sistema abierto de interconección (Open System Interconnection). 13

PDA Asistente Digital Personal (Personal Digital Assistant). 2, 7

PHY capa física del modelo OSI. 14

RFID identificación por radiofrecuencia (Radio Frequency IDentification). 2

SS intensidad de la señal (Siganl Strenght). 14

TCP Protocolo de control de transmisión (Transmission Control Protocol). 14

TDoA diferencial del tiempo de llegada (Time Diference of Arrival). 14

ToA tiempo de llegada (Time of Arrival). 14

WCDMA Acceso múltiple por división de código de banda ancha (Wideband Code Division Multiple Access). 2

Wi-Fi Tecnología de comunicación inalámbrica de datos, empleada en redes de área local. 2, 3, 7, 13

WLAN Red de área local inalámbrica (Wireless Local Area Network). 2, 3, 14

Bibliografía

[1] Diario, *La Nación*, sección de Tecnología, publicado el 5 de noviembre de 2007, Argentina

[2] Página oficial de la Comisión Federal de Telecomunicaciones, http://www.cofetel.gob.mx/es/Cofetel_2008/Cofe_servicios_de_internet, consultada el día 7 de septiembre del 2010

[3] Beresford A.R. y F. Stanjano, *Location Privacy in Pervasive Computing*, en la revista IEEE Pervasive Computing, 2003, IEEE p. 46-55

[4] Andreas Go$_r lach$, $A. Heinemann and W. W. Terpstra$
Survey on Location Privacy in Pervasive Computing

[5] Duckham, M., and L. Kulik, *Location Privacy and Location-aware computing, in Dynamic & Mobile GIS: Investigating Change in Space and Time*, J. Drummond. Et al.Editors 2006. CRC impreso en Boca Raton, FL USA p. 34-51.

[6] Marmasse, N. Y C. Schmant's, *Location-aware information delivery with comMotion*, In HUC 2K: 2nd International Symposium on Handheld and Ubicuitous Computing 2000 bristol, UK: Springer.

[7] Hariharan R. Y K. Toyama, *Project lachesis: Parsing and Modeling Location Histories*, tercera conferencia conferencia internacional de GIScience 2004, Adelphi, MD.

[8] Hightower J., et al, *Learning and recognizing the places we go*, en UbiCamp 2005, Ubicuitous Computing 2005.

[9] Hoh B., *Enhancing security and privacy in traffic monitoring systems*, IEEE Pervasive Computing Magazine 2006. p. 38-46

[10] Krumm J., *Interference Attacks on Location Tracks*, in Fifth International Conference on Pervasive Computing (pervasive 2007), 2007 Toronto, Ontario. Canada p.127-143.

[11] Gruteser M. Y B. Hoh, *On the anonimity of periodic location samples*, in the second International Conference on Security and Pervasive Computing, Boppard, Germany 2005. p. 179-192

[12] Blackman S., *Multiple-Target Tracking with Radar Applications*, 1986 Artech House

[13] Wilson D. Y C. Atkenson, *Simultaneous Tracking & Activity Recognition (STAR) Using many Anonymous, Binary Sensors*, in the third International Conference on Pervasive Computing 2005, Springer: Munich, Germany p.62-79

[14] Duckham M., L. Kulik y A. Birtley, *A spatiotemporal model of strategies and counter strategies for location privacy protection*, in 4th International Conference on Geographic Information Science (GIScience 2006), sprinter: Münster Germany, 2006 p47-64

[15] Gruteser M y D. Grunwald, *Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloacking*, in the first ACM/USENIX Internation Conference on Mobile Systems, Applications and Services, (MobySys 2003) San Francisco USA, 2003

[16] Bettini. C., X. S. Wang y S. Jajodia, *Protecting Privacy Againts Location-Based Personal Idetification*, in the 2nd VLDB Workshop on Secure Data Management, 2005.

[17] Duckham M. Y L. Kulik, *A formal model of obfuscation and negotiation for location privacy*, in the 3rd International Conference on Pervasive Computing, Springer Münich Germany, 2005, p. 152-170.

[18] Krumm J., *Inference Attacks on Location Tracks*, in the fifth International Conference on Pervassive Computing 2007, Toronto, Ontario Canada. p. 127-143

[19] Hoh B. Y M. Gruteser, *Protecting Location Privacy Through Path Confusion*, in the first International Conference on Security and Privacy for Emerging Areas in communications Networks, 2005, Athens Greece, p.194-205

[20] Tao Jiang, Helen J. Wang, y Yih-Chun Hu, *Preserving Location Privacy in Wireless LANs*, en MobySys 07, San Juan, Puerto Rico USA 2007.

[21] Ming Lei, Xiaoyan Hong y Susan Vrbsky, *Protecting Location Privacy with Dinamyc Mac Address Exchanging in Wireless Networks*, IEEE GLOBECOM 2007.

[22] Tun-Hao You, Wen-Chih Peng y Wang-Chien Lee, *Protecting Moving Trajectories with Dummies*, IEEE 2007.

[23] Santosh Pandey y Prathima Agrawal, *A survey on Localization Techniques for wireless networks*,

[24] Javier Gomez, Marco Gonzalez, Miguel Lopez, Jose Torres y Victor Rangel, *GUIDE: Guiding Users in Distributed Enviroments for WLAN and Ad-HocNetworks*,