

Universidad de Alcalá
Escuela Politécnica Superior

MÁSTER UNIVERSITARIO EN INGENIERÍA DE
TELECOMUNICACIÓN



Trabajo Fin de Máster

ESTUDIO DE UN SISTEMA DE POSICIONAMIENTO 3D DE DRONES



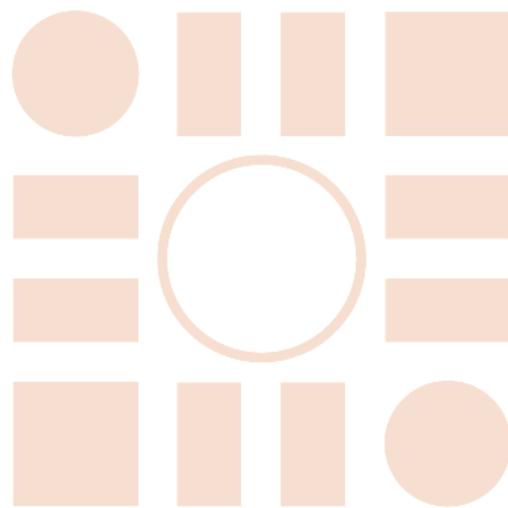
ESCUELA POLITECNICA
SUPERIOR

Autor: EVA LORETO TUDANCA CAPÓN

Tutor: JESÚS UREÑA UREÑA

2019

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá

UNIVERSIDAD DE ALCALÁ

Escuela Politécnica Superior

INGENIERÍA DE TELECOMUNICACIÓN

Trabajo Fin de Máster

Estudio de un sistema de posicionamiento 3D de drones

Autor: Eva Loreto Tudanca Capón

Tutor: Jesús Ureña Ureña

TRIBUNAL:

Presidente: Elena López Guillén

Vocal 1o: Pilar Jarabo Amores

Vocal 2o: Jesús Ureña Ureña

FECHA:

I. ÍNDICE

I.	ÍNDICE	iii
I.I	Índice de figuras	v
I.II	Índice de tablas	viii
II.	RESUMEN EN CASTELLANO	1
III.	RESUMEN EN INGLES	2
IV.	PALABRAS CLAVE.....	3
V.	RESUMEN EXTENDIDO	4
VI.	MEMORIA.....	6
1	INTRODUCCIÓN	6
2	LOCALIZACIÓN 3D BASADA EN DISTANCIA O DIFERENCIA DE DISTANCIA	8
2.1	Posicionamiento por linealización	9
2.1.1	Trilateración esférica	9
2.1.2	Trilateración hiperbólica	11
2.2	Posicionamiento Gauss-Newton	12
2.2.1	Trilateración esférica	13
2.2.2	Trilateración hiperbólica	15
3	ANÁLISIS DEL PDOP	18
3.1	PDOP mediante métodos heurísticos	18
3.2	PDOP mediante la inversa de la matriz Jacobiana	19
3.3	PDOP mediante el cálculo de la altura de tetraedros.....	20
3.3.1	Bideterminante Cayley-Menger.....	21
4	ESQUEMA PARA SIMULACIÓN 3LPS.....	26
4.1	Modo individual/independiente	28
4.2	Modo simultáneo	29
5	ALGORITMO DE FUSIÓN.....	30
6	RESULTADOS DE SIMULACIÓN	32
6.1	Modo independiente	32
6.1.1	Trilateración esférica	32
6.1.2	Trilateración hiperbólica	41
6.1.3	PDOP de modo independiente	48
6.1.3.1	PDOP mediante métodos heurísticos	48
6.1.3.2	PDOP mediante la inversa de la matriz Jacobiana	50
6.1.3.3	PDOP mediante el cálculo de la altura de tetraedros.....	52
6.1.3.4	PDOP mediante el Bideterminante de Cayley-Menger	53
6.2	Algoritmo de fusión.....	54

6.2.1 PDOP de fusión	59
6.3 Modo simultáneo	60
6.3.1 Trilateración esférica	61
6.3.2 Trilateración hiperbólica	69
6.3.3 Pérdida de balizas	75
6.3.4 PDOP de modo simultáneo	81
6.3.4.1 PDOP mediante métodos heurísticos	81
6.3.4.2 PDOP mediante la inversa de la matriz Jacobiana	83
7 CONCLUSIONES	86
VII. DIAGRAMA SW	88
VIII. PLIEGO DE CONDICIONES.....	92
IX. PRESUPUESTO	93
X. MANUAL DE USUARIO	95
XI. BIBLIOGRAFÍA.....	98

I.1 Índice de figuras

Ilustración 1. Tecnologías de posicionamiento [11].	6
Ilustración 2. Ejemplo de cálculo de Trilateración para la obtención de un móvil [12].	8
Ilustración 3. Esquema de técnicas a utilizar.	9
Ilustración 4. Ejemplo de diferencia de distancias para localización hiperbólica.	11
Ilustración 5. Diagrama de bloques para el método de Gauss-Newton.	15
Ilustración 6. Ejemplo de cálculo de la altura de tetraedros con la esfera de radio unidad [2].	21
Ilustración 7. Esquema básico del problema de la trilateración [4].	23
Ilustración 8. Cálculo de nuevas balizas con distancia de radio unidad desde el móvil.	25
Ilustración 9. Sistema de balizas a utilizar [1].	26
Ilustración 10. Esquema del entorno de simulación con un conjunto de balizas.	27
Ilustración 11. Posición de las balizas.	27
Ilustración 12. Esquema del entorno de simulación con tres conjuntos de balizas.	28
Ilustración 13. Rotación de los ejes para posicionar los distintos sistemas de balizas.	29
Ilustración 14. Posicionamiento en 3D con el método Gauss-Newton esférico.	33
Ilustración 15. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=0$).	34
Ilustración 16. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=2$).	35
Ilustración 17. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=4$).	36
Ilustración 18. Proyecciones en todos los planos de la Ilustración 14.	37
Ilustración 19. Ejemplo de elipsoides de error y proyecciones en el plano ' $z=0$ ' de la Ilustración 14.	38
Ilustración 20. Función CDF que representa el error de los 9 puntos que se sitúan en el plano ' $z=2$ ' (Gauss-Newton esférico).	39
Ilustración 21. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton esférico).	39
Ilustración 22. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton esférico).	40
Ilustración 23. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico).	41
Ilustración 24. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=0$).	42
Ilustración 25. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=2$).	43
Ilustración 26. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=4$).	44
Ilustración 27. Ejemplo de elipsoides de error y sus proyecciones del método Gauss-Newton hiperbólico.	45
Ilustración 28. Función CDF que representa el error de los 9 puntos que se sitúan en el plano ' $z=2$ ' (Gauss-Newton hiperbólico).	45
Ilustración 29. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton hiperbólico).	46
Ilustración 30. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).	47
Ilustración 31. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).	48
Ilustración 32. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).	49
Ilustración 33. PDOP del posicionamiento hiperbólico mediante métodos heurísticos ($z=2$).	50
Ilustración 34. PDOP del posicionamiento esférico mediante la inversa de la matriz Jacobiana ($z=2$).	51
Ilustración 35. PDOP del posicionamiento hiperbólico mediante la inversa de la matriz Jacobiana ($z=2$).	51

Ilustración 36. PDOP de posicionamiento mediante el cálculo de la altura de tetraedros ($z=2$).	52
Ilustración 37. PDOP de posicionamiento mediante el Bideterminante de Cayley-Menger ($z=2$).	53
Ilustración 38. Posicionamiento en 3D con el método Gauss-Newton esférico a través del uso de 3 sistemas de referencia distintos.	54
Ilustración 39. Posicionamiento 3D a través del método de fusión MLE.	55
Ilustración 40. Proyecciones en todos los planos de la Ilustración 39.	56
Ilustración 41. Elipsoide de error y proyecciones en el plano 'z' de la Ilustración 40.	56
Ilustración 42. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Fusión).	57
Ilustración 43. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Fusión).	58
Ilustración 44. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Fusión).	58
Ilustración 45. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Fusión).	59
Ilustración 46. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).	60
Ilustración 47. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=0$).	61
Ilustración 48. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=2$).	62
Ilustración 49. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=4$).	62
Ilustración 50. Proyecciones en todos los planos de la Ilustración 47, 48 y 49.	63
Ilustración 51. Ejemplo de elipsoide de error y proyecciones en el plano 'z=0' de la Ilustración 47, 48 y 49.	64
Ilustración 52. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton esférico).	65
Ilustración 53. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton esférico).	66
Ilustración 54. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton esférico).	67
Ilustración 55. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico). Parte I.	68
Ilustración 56. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico). Parte II.	69
Ilustración 57. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=0$).	70
Ilustración 58. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=2$).	70
Ilustración 59. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=4$).	71
Ilustración 60. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton hiperbólico).	72
Ilustración 61. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).	73
Ilustración 62. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton hiperbólico).	74
Ilustración 63. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).	75
Ilustración 64. Función CDF que representa el error de todos los puntos situados en un mismo plano ($z=2$) para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton esférico).	76

Ilustración 65. Error de posición de 10 cm de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton esférico).....	77
Ilustración 66. Error de posición de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas con una probabilidad mayor del 90% (Gauss-Newton esférico).....	78
Ilustración 67. Función CDF que representa el error de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton hiperbólico).	79
Ilustración 68. Error de posición de 0,5 m de todos los puntos situados en un mismo plano ($z=2$) para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton hiperbólico).	80
Ilustración 69. Error de posición de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas con una probabilidad mayor del 90% (Gauss-Newton hiperbólico)	81
Ilustración 70. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).	82
Ilustración 71. PDOP del posicionamiento hiperbólico mediante métodos heurísticos ($z=2$). ...	83
Ilustración 72. PDOP del posicionamiento esférico mediante la inversa de la matriz Jacobiana ($z=2$).....	84
Ilustración 73. PDOP del posicionamiento hiperbólico mediante la inversa de la matriz Jacobiana ($z=2$).....	85
Ilustración 74. Diagrama de SW genérico.....	88
Ilustración 75. Diagrama SW de los scripts referentes a un único conjunto de balizas y a 3 sistemas de balizas.....	89
Ilustración 76. Diagrama SW del script que intenta replicar el posicionamiento de 3 conjuntos de balizas a través de la fusión de datos.	90
Ilustración 77. Diagrama SW para los scripts de cálculo del PDOP.	91
Ilustración 78. Directorio de programación.....	95
Ilustración 79. Ventana del directorio.	96
Ilustración 80. Path para el directorio de programación.....	96

I.II Índice de tablas

Tabla 1. Costes directos.	93
Tabla 2. Costes indirectos.	94
Tabla 3. Costes del proyecto.	94

II.RESUMEN EN CASTELLANO

Hoy en día, la necesidad de localización y navegación para personas, móviles u objetos, en entornos interiores o exteriores va tomando cada vez más importancia, debido a los avances tecnológicos que lo posibilitan y a las aplicaciones que de ello se derivan. En espacios interiores no existe una solución única (como GNSS en exteriores). Normalmente se presenta una solución en base a Sistemas de Posicionamiento Local (LPS) que puede comprender posicionamiento 2D o 3D, según necesidades. Este trabajo propone el estudio de un sistema LPS 3D donde se analizan los distintos métodos para obtener la posición cuantificando los errores cometidos.

III.RESUMEN EN INGLES

Nowadays, the need for location and navigation of people, vehicles or objects, in indoor or outdoor environments is becoming increasingly important, due to the technological advances that make it possible and the applications that can be derived. In indoor spaces there is no single solution (such as GNSS outdoors). Normally there is proposed a solution based on Local Positioning Systems (LPS) that can include 2D or 3D positioning, according to needs. This work proposes the study of a 3D LPS system where different methods are analysed to obtain the position with a quantification of errors.

IV. PALABRAS CLAVE

Posicionamiento 3D LPS

Trilateración esférica e hiperbólica

Posicionamiento Gauss-Newton

Fusión de datos MLE

PDOP

V.RESUMEN EXTENDIDO

Históricamente, la navegación ha sido una necesidad y un método que ha permitido a la humanidad crecer y evolucionar hacia una sociedad global en la que las fronteras se han acercado tanto como se las conoce actualmente.

Como ya se sabe, la navegación en entornos exteriores para poder posicionar personas y objetos que estén o no en movimiento, y guiarlos en los trayectos que realizan con exactitud y simplicidad ha sido el principal objetivo hasta hace relativamente poco tiempo. Esta navegación se basa en el ampliamente conocido sistema GNSS (como el precursor GPS y los posteriores Galileo, Glonass y Beidou) desarrollado inicialmente por la industria militar, ampliamente probado y cuyo uso está totalmente establecido en todos los estratos de la sociedad, pero ¿qué pasa en entornos interiores?

Las nuevas tecnologías, como por ejemplo la domótica o la robótica, están creando nuevas necesidades de navegación que conviene estudiar para crear las soluciones adecuadas. Estos entornos interiores tienen características radicalmente distintas a las que se pueden encontrar en la navegación exterior: se necesita mayor precisión, el sistema GPS requiere cobertura de satélite que puede ser bloqueada parcial o totalmente en los entornos interiores, hay una mayor presencia de obstáculos, etc. En estos entornos, los sistemas de navegación utilizan normalmente un posicionamiento en dos dimensiones. Estos sistemas tratan de calcular la posición de un objeto o móvil implementando un Sistema de Posicionamiento Lineal (LPS) a partir de distintos algoritmos que usan la distancia con varios puntos de referencia, generalmente balizas emisoras, receptoras o emisoras/receptoras.

El avance de la robótica, más concretamente con vehículos no tripulados como drones cambian los requisitos de la navegación en interiores. Los primeros sistemas para posicionamiento interior tenían como objetivo la situación de objetos y/o personas dentro de un edificio. Por ejemplo, en un congreso los asistentes podrían guiarse usando su móvil para situarse en la habitación o sala del edificio en la que se encuentren y desde ahí guiarse para llegar a cualquier otro lugar. Sin embargo, un dron además de situarse en las dos dimensiones en las que se sitúa una persona, necesita conocer la altura a la que se encuentra. Viendo la acuciante necesidad de este posicionamiento en interiores, se empiezan a desarrollar sistemas de posicionamiento en 3D para ampliar el uso de los sistemas LPS, y que permiten un amplio abanico en cuanto a desarrollo de nuevas aplicaciones, ya que el empleo de drones está ahora mismo en auge impulsando a su vez el desarrollo de los sistemas LPS de tres dimensiones.

Durante los últimos diez años, se han realizado avances cuantificables en el desarrollo de estos sistemas de posicionamiento en interiores, que unido al creciente uso de drones y la aparición de nuevas tecnologías y aplicaciones han impulsado el desarrollo y la investigación de los sistemas LPS. Es por ello también que se desarrollan a la par una gran cantidad de productos que buscan alcanzar ese nivel de precisión que está buscando satisfacer las necesidades de un mercado con un crecimiento constante y unas altas expectativas.

Los sistemas LPS basan mayoritariamente su funcionamiento en balizas que se encuentran situadas en el entorno. Los objetos a localizar, o que quieran conocer su posición absoluta o relativa, usarán transmisores o receptores para detectar las señales de las balizas, que utilizarán para calcular métricas, principalmente la distancia, como entradas de distintos algoritmos de posicionamiento. Es por ello, que en la última década se ha evolucionado de manera conjunta tanto en los sistemas de comunicaciones inalámbricos como los sistemas LPS.

Consecuentemente, el desarrollo de los sistemas LPS de tres dimensiones conlleva una investigación centrada principalmente en la mejora de los algoritmos de posicionamiento, la estimación de la posición de objetos, como puede ser un dron, a partir de la fusión de datos provenientes de diferentes sensores, algoritmos para el tratamiento y procesado de datos provenientes de distintas fuentes, tratamiento de las imágenes en caso de disponer de cámaras, etc.

En este Trabajo Fin de Máster, se desarrollan y se realizan simulaciones de varios algoritmos de posicionamiento para estimar el resultado, en forma de posición, y la precisión en un entorno de dimensiones acotadas; y obtener una conclusión sobre la idoneidad y condiciones de uso de cada uno de ellos.

En primer lugar, se analizarán las distintas maneras de posicionamiento que se conocen para observar cómo funcionan los distintos algoritmos y cómo se va a desarrollar su implementación. A saber, se hablará del posicionamiento por linealización del sistema de ecuaciones no lineales y del posicionamiento por resolución de dicho sistema usando el método de Gauss-Newton.

Una vez realizada la implementación de los diferentes algoritmos de posicionamiento, se tiene que cuantificar la precisión de los resultados obtenidos. Para ello, se debe encontrar un método que sea capaz de cuantificar el error de los distintos algoritmos usados para calcular la posición. Este método es el conocido como PDOP (Precision of Dilution of Position), que permite cuantificar el error de posicionamiento, causado por los errores cometidos al medir las distancias.

El proceso de simulación para realizar las pruebas del funcionamiento de los distintos algoritmos se realizará de la siguiente forma:

Se definirá un entorno de determinadas características para poder representar el posicionamiento de un determinado móvil que se irá moviendo a lo largo de ese entorno. En dicho entorno se situará un sistema de referencia compuesto por varias balizas que permitirán realizar las medidas de distancias o distancias relativas que necesitan los algoritmos para estimar la posición del objeto. El objeto se irá moviendo a lo largo de los tres planos espaciales, estimando su posición en un número definido de puntos. Una vez estimada la posición en esos puntos, se cuantificará la precisión del algoritmo.

Las simulaciones de posicionamiento se realizan de dos maneras distintas: modo independiente y modo simultáneo. En el modo independiente, se dispone en el área de trabajo de un único sistema de referencia o de varios sistemas de referencia, mientras que en el modo simultáneo se dispondrá de varios sistemas de referencia. En el primer caso, antes de poder realizar la estimación de la posición, se requiere fusionar los datos medidos por los diversos sistemas de balizas para obtener la posición del objeto y poder comunicarla de forma que sea útil en un sistema de navegación. En este trabajo se propone un método para fusionar los datos obtenidos de diversos sistemas de referencia y obtener un único resultado.

Finalmente, se presentan los resultados obtenidos de las distintas simulaciones realizadas, además de las conclusiones que se pueden obtener de los resultados obtenidos.

En este trabajo, también se realiza una estimación de los costes del proyecto realizado.

Se incluye también un manual de usuario y se propone un escenario que permita replicar y continuar con la investigación propuesta.

VI. MEMORIA

1 INTRODUCCIÓN

La navegación y el posicionamiento se han expandido gracias a las nuevas tecnologías y, actualmente, además de su aplicación en entornos exteriores destaca su creciente interés en las aplicaciones en entornos interiores.

Los Sistemas de Posicionamiento Local (LPS) tienen el objetivo de detectar, posicionar e incluso seguir objetos en movimiento en un entorno interior, donde un sistema no tiene cobertura o no proporciona la precisión necesaria para la aplicación deseada. Estos sistemas LPS usan distintas tecnologías dependiendo de las necesidades finales o de la arquitectura que se encuentre desplegada en el entorno, como los sistemas de ondas ópticas, acústicas, infrarrojos o los basados en radiofrecuencia.

Estas tecnologías son ampliamente utilizadas debido al crecimiento, que se ha experimentado en los últimos años, de los sistemas inalámbricos; generando unas altas expectativas tales como una precisión en el orden de los decímetros. En la Ilustración 1 se puede ver el amplio uso que tienen hoy en día los sistemas comentados anteriormente, al igual que las precisiones que pueden ofrecer.

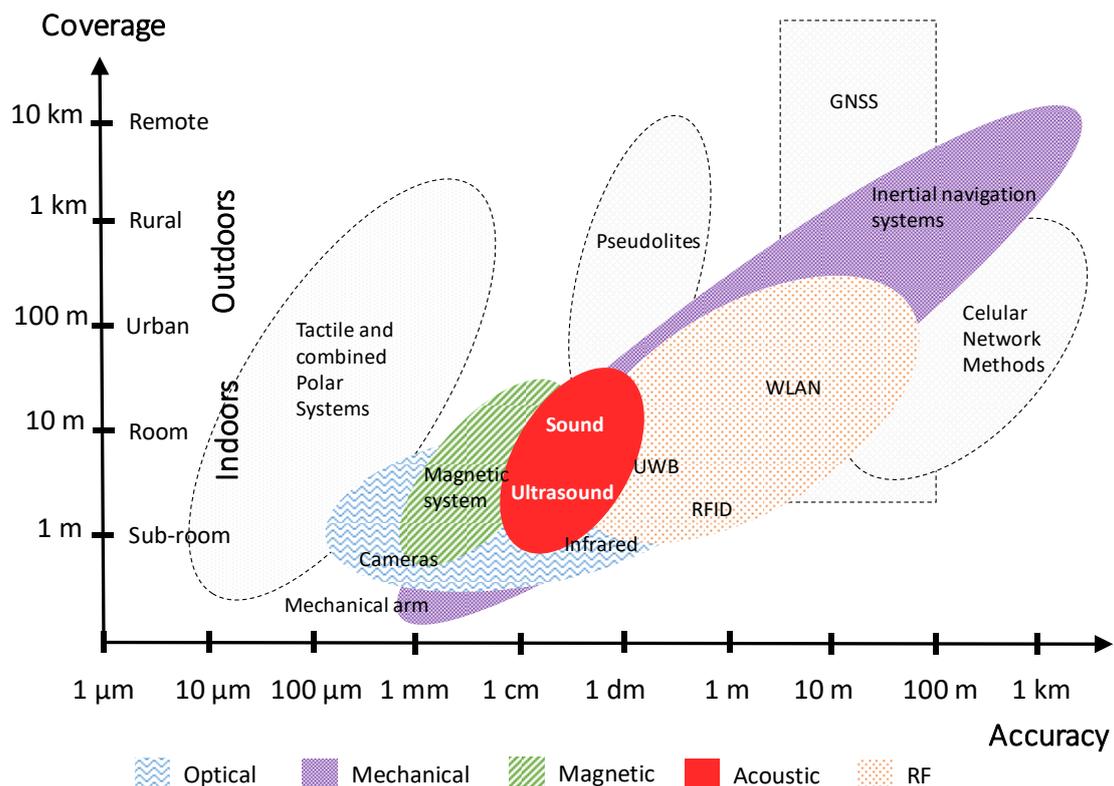


Ilustración 1. Tecnologías de posicionamiento [11].

En este proyecto, de las distintas tecnologías que se utilizan en los sistemas LPS, se va a estudiar un sistema de posicionamiento en tres dimensiones basado en ultrasonidos. La elección de esta tecnología en particular se debe al hecho de que ha mostrado buenos resultados, logrando precisiones en el rango de centímetros. Además, es una tecnología asequible económicamente y los objetos de estudio, los drones, se moverán a una velocidad relativamente baja al estar

confinados en un espacio relativamente pequeño, en el cual no podrán moverse a las altas velocidades que pueden alcanzar. La tecnología ultrasónica ya ha sido usada para posicionar robots, demostrando unas prestaciones acordes a lo que el mercado reclama [\[8\]](#) [\[9\]](#). De hecho, hay sistemas comerciales, como los robots aspiradores, por ejemplo, que utilizan tecnología ultrasónica para localizar y volver a la base. Por todas estas razones, los sistemas de ultrasonidos son los más adecuados para realizar el estudio sobre los algoritmos de posición.

El estudio de los algoritmos analizará la distribución de las balizas en un entorno de tres dimensiones limitado en la totalidad de su volumen (tipo habitación) para garantizar una cobertura y un PDOP (Precision of Dilution of Position) por debajo de un determinado valor, es decir, una precisión mínima que garantice el cumplimiento de los requisitos que vienen determinados por la demanda de las aplicaciones que se implementan.

Una vez realizadas las simulaciones y con los resultados obtenidos, se podrá proponer una implementación práctica del sistema basado en balizas ultrasónicas. El objetivo final de este sistema es, además de ser lo suficientemente preciso para posicionar el objeto móvil de acuerdo con los requisitos que imponen los usos prácticos, que sea económico tanto para el proceso de desarrollo como para los clientes finales.

2 LOCALIZACIÓN 3D BASADA EN DISTANCIA O DIFERENCIA DE DISTANCIA

La idea de localización es ser capaz de posicionar un móvil dentro de un entorno y ser capaz de seguir a ese móvil si este se encuentra dentro de un entorno en movimiento. Para poder lograr ese posicionamiento de un cierto móvil se tienen distintas opciones. La principal en la que se va a centrar este trabajo es en el posicionamiento por trilateración. Ello consiste en obtener una posición a través de ciertas magnitudes que se harán medibles mediante ecuaciones algebraicas. Para ello se necesita conocer ciertas variables que se van a tomar como conocidas como pueden ser en este caso un conjunto de balizas que formarán el sistema de referencia a emplear.

Mediante el uso de la trilateración, la idea es tener 'N' ecuaciones a resolver donde las variables de navegación para obtener la posición del móvil son desconocidas. Se considerarán tantas ecuaciones como tantas balizas se tengan como mínimo; 'N' ecuaciones, 'N' o más balizas.

Una vez conocido esto, para el posicionamiento se necesita obtener conocimiento de la distancia desde el blanco a localizar hasta el sistema de referencia elegido, las balizas. Otra opción es obtener una posición a través de la diferencia de distancias, así se hace uso de una distancia relativa.

En la siguiente figura se ilustra un ejemplo de lo acabado de comentar anteriormente.

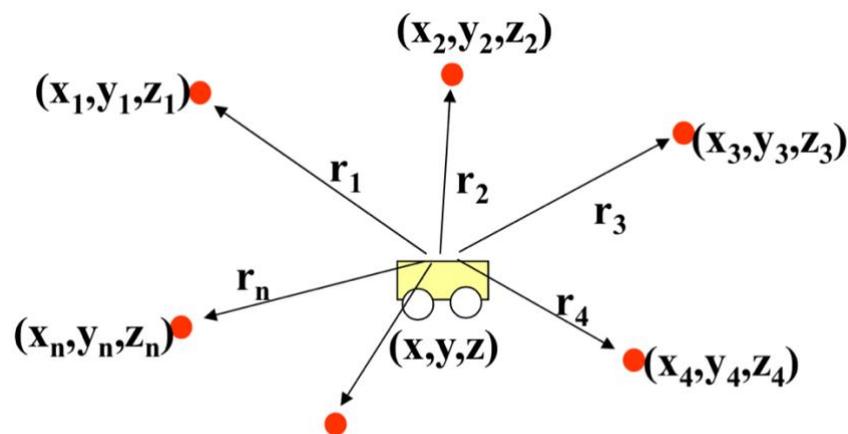


Ilustración 2. Ejemplo de cálculo de Trilateración para la obtención de un móvil [12].

Como se aprecia en la ilustración anterior, se muestra un entorno donde se sitúan varios puntos conocidos a partir de los cuales se puede medir la distancia al objeto que se quiere posicionar mediante alguna técnica como puede ser el tiempo de vuelo de la emisión ultrasónica desde la baliza al receptor en el móvil. Esos puntos conocidos son los que se conocerán como balizas, y formarán un sistema conocido para este proyecto como se verá en apartados futuros. Como las distancias de las balizas al objeto a posicionar son también conocidas (medidas), se puede formar un conjunto de ecuaciones con esas variables para poder calcular la posición estimada del objeto. Se puede entender que, cuantas más posiciones de referencia se tengan conocidas, mejor podrá ser el posicionamiento del móvil debido a que el sistema de ecuaciones será mayor al disponer de un mayor número de datos.

Por otro lado, la opción más práctica para resolver esas ecuaciones va a ser el uso de sistemas lineales de ecuaciones (LSE).

A continuación, se va a caracterizar el sistema básico del que se va a hacer uso para las simulaciones. Este sistema constará de una serie de balizas que se tendrán colocadas en unas posiciones elegidas dentro de un entorno controlado. Se sabrá dónde estará el móvil del que se va a obtener la posición, y se introducirá en el entorno una medida de error debido a que tanto en los sistemas de medida como en el propio medio siempre se genera error que no se puede eliminar. Más detalles de este sistema se analizarán en el apartado de 'Esquema para simulación'.

Teniendo ya una idea del sistema, se van a emplear dos técnicas de trilateración. La trilateración esférica e hiperbólica. La primera de ellas se basa en las distancias entre el móvil a posicionar y el sistema de balizas para poder obtener la posición de ese móvil, una posición estimada de la cual se intentará reducir al máximo el error de posicionamiento. En cambio, la segunda opción se basa en las diferencias de distancias entre el punto donde se sitúa el móvil y el sistema de balizas, en relación a una de ellas tomada como referencia.

A partir de esas dos técnicas (esférica e hiperbólica), se procede a resolver los sistemas de ecuaciones mencionados anteriormente mediante el empleo de dos soluciones, ambas para cada técnica. Estas soluciones son la linealización de los sistemas de ecuaciones a través de una baliza adicional o usando el método de Gauss-Newton.

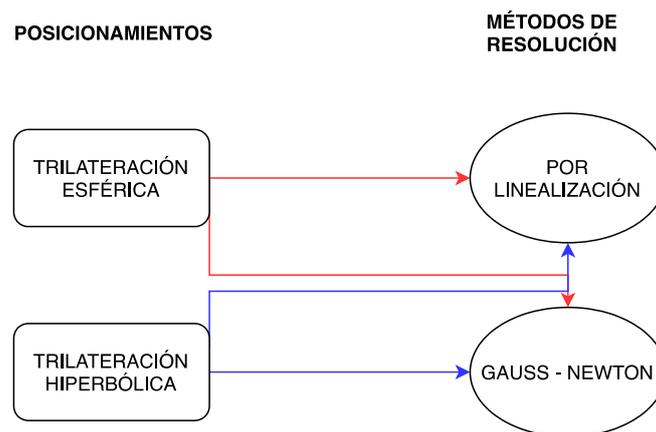


Ilustración 3. Esquema de técnicas a utilizar.

2.1 Posicionamiento por linealización

Esta es una técnica donde se va a resolver el sistema de ecuaciones generado por el sistema de balizas que se tenga, y por los puntos donde se va a posicionar el objeto móvil. Se procederá a verificar que con esta técnica se pueden resolver ambas maneras de trilateración del objeto de estudio simplemente cambiando la manera en la que se recogen las distancias como bien se ha comentado anteriormente. Después de analizar este estudio que se hará a continuación, se comprenderá mejor lo comentado.

Se va a empezar a analizar la técnica de posicionamiento por linealización sobre el método de trilateración esférica para luego ver cómo se aplica lo mismo para la trilateración hiperbólica.

2.1.1 Trilateración esférica

Se tiene que conocer donde está situado el sistema de referencia, las balizas, y haber calculado la distancia que hay entre el objeto, situado en un punto (x, y, z) del entorno donde se va a

mover, y el sistema de balizas. Para poder situar este objeto en 3D se necesitan como mínimo la distancia de 4 balizas para poder posicionarlo adecuadamente. La Ilustración 2 muestra un ejemplo de nomenclatura para trilateración esférica (que se va a seguir a continuación) donde se pueden ver las distancias desde una posición conocida al objeto móvil del que se va a calcular su posición.

Teniendo estos datos, se pueden calcular las matrices conocidas como **A** y **B** que surgen del agrupamiento de los distintos sumandos que componen el cálculo de la distancia como se muestra a continuación:

- Cálculo de la distancia: ecuación de la esfera con centro en la posición de cada baliza "i" y de radio la distancia medida, r_i):

$$\begin{aligned} r_i^2 &= (x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 \\ &= x^2 + y^2 + z^2 + x_i^2 + y_i^2 + z_i^2 - 2xx_i - 2yy_i - 2zz_i \end{aligned} \quad (\text{Ec. 1})$$

- Agrupación de sumandos, con la inclusión de una variable adicional ($t=x^2+y^2+z^2$) y una constante adicional ($S_i^2= x_i^2+y_i^2+z_i^2$), para construir el sistema de ecuaciones para obtener el posicionamiento del objeto conociendo la distancia del objeto a las balizas:

$$x^2 + y^2 + z^2 + x_i^2 + y_i^2 + z_i^2 - 2xx_i - 2yy_i - 2zz_i - r_i^2 = 0 \quad i = 1, 2, \dots, n$$

$$t + S_i^2 - 2xx_i - 2yy_i - 2zz_i - r_i^2 = 0 \quad i = 1, 2, \dots, n$$

$$\mathbf{X} = \begin{pmatrix} x \\ y \\ z \\ t \end{pmatrix}; \mathbf{A} = \begin{pmatrix} 2x_1 & 2y_1 & 2z_1 & -1 \\ 2x_2 & 2y_2 & 2z_2 & -1 \\ \dots & \dots & \dots & \dots \\ 2x_n & 2y_n & 2z_n & -1 \end{pmatrix}; \mathbf{B} = \begin{pmatrix} S_1^2 - r_1^2 \\ S_2^2 - r_2^2 \\ \dots \\ S_n^2 - r_n^2 \end{pmatrix}$$

(Ec. 2)

- Sistema de ecuaciones a resolver:

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B}$$

(Ec. 3)

A es una matriz que guarda la información de las balizas añadiéndole una adicional, la columna de todo unos, que puede estar situada en cualquier sitio; y **B** es una matriz donde se combinan la posición de las balizas con las distancias medidas desde las balizas al punto objetivo.

Con este sistema generado, se logra obtener una posición estimada del blanco con un determinado error, que dependerá de la precisión de las medidas y de la distribución de las balizas. Este sistema puede tener múltiples singularidades causadas por el sistema de referencia

de balizas. Una de esas singularidades se puede deber a que se tengan las balizas coplanares debido a su situación. Si se llega a un punto donde se cumpliera esto, habría que ver cuál sería la mejor manera de resolver esta singularidad. En este caso se comprobará que el sistema de baliza no es coplanar en puntos posteriores.

2.1.2 Trilateración hiperbólica

Otra forma de calcular la posición del objeto es a través del otro método comentado, la trilateración hiperbólica. Para este método habría que seguir los mismos pasos comentados anteriormente, pero utilizando diferencias de distancias, tomando una baliza como referencia. Habría que tener en cuenta también que ahora se necesitan un mínimo de 5 balizas para posicionar el blanco en 3D.

La diferencia de distancia se obtiene a partir de la diferencia entre la distancia de una baliza al móvil y la de otra baliza de referencia al mismo móvil. Esto permite calcular la posición de un móvil sin conocer la distancia directa entre éste y el sistema de referencia del que se disponga (lo que en la práctica se puede medir sin necesidad de sincronizar el receptor al sistema de balizamiento). En la siguiente figura se muestra un ejemplo de esta distancia para tener más claro lo que se pretende expresar.

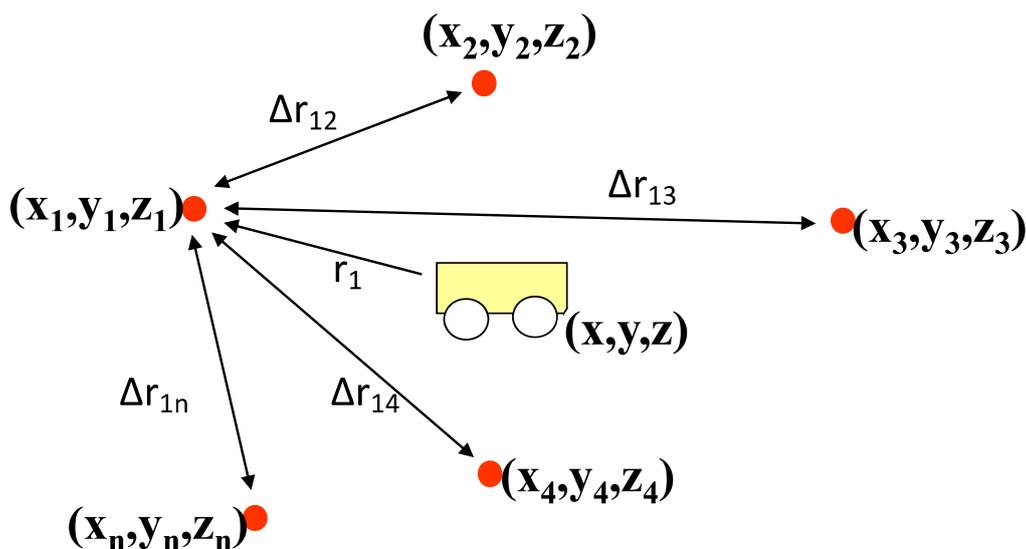


Ilustración 4. Ejemplo de diferencia de distancias para localización hiperbólica.

Cogiendo la baliza 1 como la baliza de referencia (lo que no resta generalidad), se obtienen las siguientes ecuaciones:

- Distancia del objeto a la baliza número 1 (esfera con radio r_1 y centrada en la posición de la baliza 1):

$$(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = r_1^2$$

(Ec. 4)

- Cálculo de las diferencias de distancias del resto de balizas as cogiendo la baliza 1 como la baliza referente:

$$(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = [r_1 + \Delta r_{21}]^2$$

$$(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = [r_1 + \Delta r_{31}]^2$$

$$(x - x_n)^2 + (y - y_n)^2 + (z - z_n)^2 = [r_1 + \Delta r_{n1}]^2$$

(Ec. 5)

- Agrupación de sumandos para construir el sistema de ecuaciones:

$$\mathbf{X} = \begin{pmatrix} x \\ y \\ z \\ r_1 \end{pmatrix}; \mathbf{A} = \begin{pmatrix} 2(x_2 - x_1) & 2(y_2 - y_1) & 2(z_2 - z_1) & -2\Delta r_{21} \\ 2(x_3 - x_1) & 2(y_3 - y_1) & 2(z_3 - z_1) & -2\Delta r_{31} \\ \dots & \dots & \dots & \dots \\ 2(x_n - x_1) & 2(y_n - y_1) & 2(z_n - z_1) & -2\Delta r_{n1} \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} -\Delta r_{21}^2 - x_1^2 + y_1^2 + z_1^2 + x_2^2 + y_2^2 + z_2^2 \\ -\Delta r_{31}^2 - x_1^2 + y_1^2 + z_1^2 + x_3^2 + y_3^2 + z_3^2 \\ \dots \\ -\Delta r_{n1}^2 - x_1^2 + y_1^2 + z_1^2 + x_n^2 + y_n^2 + z_n^2 \end{pmatrix}$$

(Ec. 6)

- Sistema de ecuaciones a resolver:

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B}$$

(Ec. 7)

Como se puede comprobar, el método de resolución es muy parecido, solo hay que tener en cuenta el manejo de las distancias o diferencias de distancias.

2.2 Posicionamiento Gauss-Newton

Este otro método utilizado para este estudio es una técnica donde se va a resolver el mismo dilema planteado anteriormente para el posicionamiento. Se va a volver a verificar que se pueden resolver ambas maneras de trilateración del objeto móvil simplemente cambiando la manera en la que se recogen las distancias.

Se comenzará analizando la técnica de posicionamiento Gauss-Newton sobre el método de trilateración esférica para luego ver cómo se aplica lo mismo para la trilateración hiperbólica.

2.2.1 Trilateración esférica

Sabiendo donde está situado el sistema de referencia, las balizas, y conociendo la distancia que hay entre el objeto y el sistema de balizas (medida), se puede situar el móvil con un número mínimo de 3 balizas para que el posicionamiento sea en 3D. En este caso no es necesario una baliza adicional para linealizar el sistema de ecuaciones; no obstante, cuantas más ecuaciones se tengan mejor será el resultado ante medidas afectadas por ruido.

Conociendo los datos anteriores, se puede resolver la misma ecuación calculada anteriormente, Ecuación 3 y 7, a través de la minimización de una función de coste que contiene la distancia al móvil "real" conocida y la distancia estimada que se irá calculando de manera iterativa. Esta distancia estimada es la que hace que una función de coste se vaya minimizando para lograr obtener con un mínimo de error la posición del móvil. Como posición estimada inicial se debe recurrir al conocimiento apriorístico que se tenga del entorno. En ese caso se inicializa el algoritmo en una posición centrada del entorno de estudio (cerca de la baliza central), y se calcula la distancia desde esa posición inicial estimada al sistema de balizas. A través de la resolución del sistema de manera iterativa, se irá obteniendo la posición del objeto que se irá reduciendo en cada iteración y acercando a la posición "real" poniendo como límite una tolerancia de error en el algoritmo.

El método seguido para obtener ahora las matrices **A** y **B** para conseguir la posición del móvil es el siguiente:

- Se parte de la distancia "real" (la distancia medida en la práctica), r_i , conocida porque se sabe donde está situado el punto a calcular, y la distancia inicial estimada, \hat{r}_i , con punto de inicio del algoritmo cerca de la baliza central:

$$\begin{aligned}(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 &= r_i^2 \\ (\hat{x} - x_i)^2 + (\hat{y} - y_i)^2 + (\hat{z} - z_i)^2 &= \hat{r}_i^2\end{aligned}$$

(Ec. 8)

- Se forma la función de coste que se va a minimizar:

$$F(x, y, z) = \sum_{i=1}^n (r_i - \hat{r}_i)^2 = \sum_{i=1}^n [f_i(x, y, z)]^2$$

(Ec. 9)

- Se le aplica a la función de coste anterior las derivadas parciales de primer orden en cada coordenada para poder definir la matriz Jacobiana **A**:

$$\frac{\delta F}{\delta x} = 2 \sum_{i=1}^n f_i \frac{\delta f_i}{\delta x}; \quad \frac{\delta F}{\delta y} = 2 \sum_{i=1}^n f_i \frac{\delta f_i}{\delta y}; \quad \frac{\delta F}{\delta z} = 2 \sum_{i=1}^n f_i \frac{\delta f_i}{\delta z}$$

$$A = 2J = 2 \begin{pmatrix} \frac{\delta f_1}{\delta x} & \frac{\delta f_1}{\delta y} & \frac{\delta f_1}{\delta z} \\ \frac{\delta f_2}{\delta x} & \frac{\delta f_2}{\delta y} & \frac{\delta f_2}{\delta z} \\ \dots & \dots & \dots \\ \frac{\delta f_n}{\delta x} & \frac{\delta f_n}{\delta y} & \frac{\delta f_n}{\delta z} \end{pmatrix}$$

$$\frac{\delta f_i}{\delta x} = \frac{(x - x_i)}{\sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}} = \frac{(x - x_i)}{f_i + \hat{r}_i}$$

(Ec. 10)

- Una vez construida la matriz **A** se termina de construir el sistema de ecuaciones para obtener la posición estimada del móvil:

$$A = \begin{pmatrix} \frac{\delta F}{\delta x} \\ \frac{\delta F}{\delta y} \\ \frac{\delta F}{\delta z} \end{pmatrix}; B = \begin{pmatrix} f_1 \\ f_2 \\ \dots \\ f_n \end{pmatrix}$$

$$\Delta X = (A^T A)^{-1} A^T B$$

(Ec. 11)

Al tratarse del mismo sistema de ecuaciones, se vuelve a presentar la singularidad debida al sistema de balizas; pero ya se ha establecido en que se comprobaría en puntos posteriores que el sistema de balizas no es coplanar, por tanto, no se tienen singularidades a resolver.

Como se puede apreciar en la ecuación anterior, lo que se calcula realmente es un incremento de posición que indicará si la posición de la que se parte es aceptable o no. La manera de saber si la posición es aceptable es comparando ese incremento con una tolerancia de error que se impone en el sistema para conseguir que la posición final a la que se quiere llegar sea, en el mejor caso, la posición "real" del punto donde se quiere posicionar el móvil o una posición muy cercana.

En este caso, la tolerancia con la que se trata es de 10^{-10} ; por lo que se irán reduciendo los incrementos de posición que se irán calculando a lo largo de las iteraciones del algoritmo para llegar a una posición final estimada del móvil con esa tolerancia como máximo. Para el caso en que el algoritmo tuviese problemas de convergencia se establece también un número máximo

de iteraciones a considerar (si no se ha llegado al error mínimo se deben cambiar las condiciones iniciales o, en caso extremo, descartar las medidas).

Para tener más claro lo que se acaba de comentar, la siguiente ilustración muestra un diagrama de bloques del algoritmo para saber como obtener esa posición final con un error mínimo.

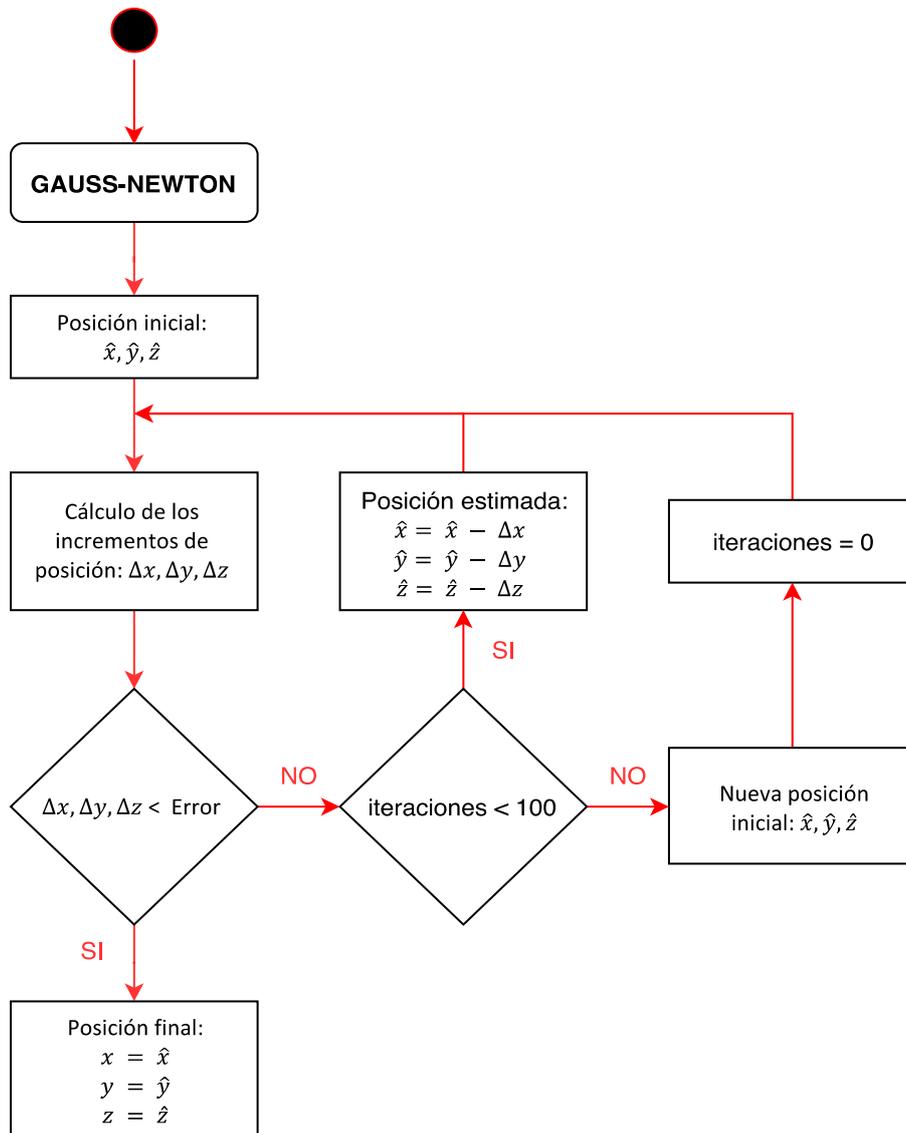


Ilustración 5. Diagrama de bloques para el método de Gauss-Newton.

2.2.2 Trilateración hiperbólica

El método de resolución para calcular la posición del móvil mediante trilateración hiperbólica es el que se va a presentar a continuación. Hay que contemplar que se parte de diferencias de distancias, por lo que se necesitará una baliza de referencia; y entonces se necesita un mínimo de 4 balizas para que el posicionamiento sea en 3D.

Cogiendo la baliza 1 como la baliza de referencia, se obtienen las siguientes ecuaciones:

- Se parte de la distancia "real" (la distancia medida en la práctica), r_i , conocida porque se sabe dónde está situado el punto a calcular, y la distancia inicial estimada, \hat{r}_i , calculada a través de la posición de inicio del algoritmo:

$$\begin{aligned}(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 &= r_i^2 \\ (\hat{x} - x_i)^2 + (\hat{y} - y_i)^2 + (\hat{z} - z_i)^2 &= \hat{r}_i^2\end{aligned}$$

(Ec. 12)

- Se calculan las diferencias de distancias, teniendo en cuenta para todos los cálculos la baliza de referencia (la número 1):

$$\begin{aligned}\Delta r_{1i} &= r_i - r_1 \\ \hat{\Delta} r_{1i} &= \hat{r}_i - \hat{r}_1\end{aligned}$$

$$\hat{\Delta} r_{1i} = \hat{r}_i - \hat{r}_1 = \sqrt{(\hat{x} - x_i)^2 + (\hat{y} - y_i)^2 + (\hat{z} - z_i)^2} - \sqrt{(\hat{x} - x_1)^2 + (\hat{y} - y_1)^2 + (\hat{z} - z_1)^2}$$

(Ec. 13)

- Se forma la función de coste que se va a minimizar:

$$F(x, y, z) = \sum_{i=2}^n (\hat{\Delta} r_{1i} - \Delta r_{1i})^2 = \sum_{i=2}^n [f_i(x, y, z)]^2; \text{ donde:}$$

$$\begin{aligned}f_i(x, y, z) &= \hat{\Delta} r_{1i} - \Delta r_{1i} \\ &= \sqrt{(\hat{x} - x_i)^2 + (\hat{y} - y_i)^2 + (\hat{z} - z_i)^2} \\ &\quad - \sqrt{(\hat{x} - x_1)^2 + (\hat{y} - y_1)^2 + (\hat{z} - z_1)^2} - \Delta r_{1i}\end{aligned}$$

(Ec. 14)

- Se le aplica a la función de coste anterior las derivadas parciales de primer orden en cada coordenada para poder definir la matriz Jacobiana **A**:

$$\frac{\delta F}{\delta x} = 2 \sum_{i=2}^n f_i \frac{\delta f_i}{\delta x}; \quad \frac{\delta F}{\delta y} = 2 \sum_{i=2}^n f_i \frac{\delta f_i}{\delta y}; \quad \frac{\delta F}{\delta z} = 2 \sum_{i=2}^n f_i \frac{\delta f_i}{\delta z}$$

$$\mathbf{A} = 2\mathbf{J} = 2 \begin{pmatrix} \frac{\delta f_1}{\delta x} & \frac{\delta f_1}{\delta y} & \frac{\delta f_1}{\delta z} \\ \frac{\delta f_2}{\delta x} & \frac{\delta f_2}{\delta y} & \frac{\delta f_2}{\delta z} \\ \dots & \dots & \dots \\ \frac{\delta f_n}{\delta x} & \frac{\delta f_n}{\delta y} & \frac{\delta f_n}{\delta z} \end{pmatrix}$$

$$\frac{\delta f_i}{\delta x} = \frac{(x - x_i)}{\sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}} - \frac{(x - x_1)}{\sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}}$$

(Ec. 15)

- Una vez construida la matriz 'A' se termina de construir el sistema de ecuaciones para obtener la posición estimada del móvil:

$$\mathbf{A} = \begin{pmatrix} \frac{\delta F}{\delta x} \\ \frac{\delta F}{\delta y} \\ \frac{\delta F}{\delta z} \end{pmatrix}; \mathbf{B} = \begin{pmatrix} f_1 \\ f_2 \\ \dots \\ f_n \end{pmatrix}$$

$$\Delta \mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B}$$

(Ec. 16)

Como se puede comprobar, el método de resolución es el mismo, pero teniendo en cuenta las diferencias de distancias.

Se puede apreciar que, al igual que en el caso anterior, lo que se calcula realmente es un incremento de posición del cual hay que valorar si éste cumple las condiciones de error que se han establecido previamente para el algoritmo. Así se conseguirá la mejor posición estimada posible que es capaz de conseguir el algoritmo con las condiciones establecidas, a través de la reducción de los incrementos de posición.

Para este caso de trilateración hiperbólica para el método de Gauss-Newton, la tolerancia que se impone al algoritmo es de 10^{-4} . Más adelante se verá por qué se produce ese cambio de tolerancias entre los algoritmos. También se incluye la excepción de un número máximo de iteraciones para el caso de problemas de convergencia.

3 ANÁLISIS DEL PDOP

La medida DOP (dilución de la precisión) es una medida que muestra la calidad de un sistema de posicionamiento, que surgió con el desarrollo del GPS; por tanto, indica la precisión con la cual se va a poder estimar la posición mediante un algoritmo de posicionamiento cualquiera debido a la relación entre la configuración de las balizas y la posición del objeto a estimar. Es por ello, que es un factor muy importante a tener en cuenta para el desarrollo de este tipo de sistemas.

Dentro de la medida DOP, se pueden distinguir varios tipos de cálculos: GDOP (Geometric Dilution of Precision), PDOP (Position Dilution of Precision), VDOP (Vertical Dilution of Precision), HDOP (Horizontal Dilution of Precision) y TDOP (Time Dilution of Precision). Para la representación de los resultados de este proyecto, se va a optar por la representación de la posición mediante el PDOP, así se podrá entender la precisión de las medidas en cuanto a posición se hable.

Una vez que se ha elegido la medida de PDOP para la representación de los cálculos, se procederá a explicar cómo se realiza el cálculo para esa medida en concreto.

3.1 PDOP mediante métodos heurísticos

El cálculo de una medida DOP se define como la relación entre el error que se comete en el posicionamiento y el error de las mediciones que se tenga en el sistema. Para el PDOP, ese error de posicionamiento se define a través del cálculo de la varianza de cada una de las coordenadas que componen la posición estimada del móvil. Por tanto, una vez que se ha obtenido esa posición estimada a través de cualquiera de los algoritmos de posicionamiento, como puede ser el algoritmo de linealización o el de Gauss-Newton, se procede a aplicar la siguiente ecuación:

$$PDOP \approx \frac{\sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2}}{\sigma_m}$$

(Ec. 17) [6]

Donde, como se ha comentado anteriormente, se requiere la varianza de cada una de las coordenadas de la posición estimada ($\sigma_x^2 + \sigma_y^2 + \sigma_z^2$), y en el denominador se hace referencia a la desviación estándar correspondiente al ruido implícito en el sistema de posicionamiento, σ_m . Ese ruido se ha considerado de 0,01m, porque es típico en sistemas ultrasónicos.

Una vez que ya se conoce cómo obtener esa medida de precisión, se calculan las medidas de PDOP a través de la obtención de distintas medidas de posición estimada sobre una misma posición real. Más concretamente, de un mismo punto situado dentro del entorno, se realizan varias medidas de estimación de posición a partir de los dos tipos de posicionamiento comentados anteriormente, y se calcula con todas esas medidas una estimación del PDOP a través de la varianza de todas las medidas estimadas obtenidas. Es por ello que esta manera de cálculo de PDOP se llamará por métodos heurísticos. Este cálculo se repite para todos los puntos donde se quiere obtener una posición estimada para, posteriormente, obtener un mapa de PDOP para saber cómo se distribuye la precisión de todos los puntos calculados en un plano.

Hay que dejar claro que esta medida de PDOP que se está comentando, es una **medida estimada** de cálculo de la precisión debido a que se están valorando las mediciones estimadas de la posición. No muestra el resultado ideal que se podría alcanzar en este sistema. Una medida más ideal de PDOP es la que se mostrará en el siguiente apartado.

Esta manera de calcular el PDOP sólo es una de muchas maneras de estimar esta medida. Para el contraste de los posteriores resultados, se va a obtener la misma medida mediante otras formas analíticas.

3.2 PDOP mediante la inversa de la matriz Jacobiana

La segunda manera empleada para el cálculo del PDOP consiste en utilizar la función inversa. Este método consiste en conseguir la medida de precisión de error a través del cálculo de la matriz \mathbf{A} de los algoritmos de posicionamiento mencionados en apartados anteriores. Aplicando la siguiente ecuación: $(\mathbf{A}^T \mathbf{A})^{-1}$, se puede obtener directamente la medida de PDOP debido a que se obtiene una matriz cuya diagonal contiene la información que se necesita.

Esta medida de PDOP es una **medida ideal** de la precisión que se puede obtener del sistema planteado debido a que se aplica directamente el algoritmo, pero a su inversa. Esto quiere decir que se realizan medidas independientes y con similares errores para cada punto de cálculo. Así se puede establecer una cota superior que sirve como objetivo a alcanzar y con el que poder comparar resultados.

A continuación, se va a justificar lo dicho anteriormente para clarificarlo. Se parte del siguiente sistema de ecuaciones ya conocido:

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{B} \quad (\text{Ec. 18})$$

Se sabe que la matriz \mathbf{A} es una matriz Jacobiana, y que la matriz \mathbf{B} es una función de diferencia de distancias, por tanto, este mismo sistema se puede caracterizar de la siguiente manera:

$$\Delta \mathbf{X} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \Delta r = \mathbf{J}^{-1} \Delta r \quad (\text{Ec. 19})$$

Si se procede a calcular la covarianza de la posición estimada, se obtiene lo siguiente:

$$\text{Cov}(\mathbf{X}) = \Sigma_{\mathbf{X}} = E\{\Delta \mathbf{X} (\Delta \mathbf{X})^T\} = E\{\mathbf{J}^{-1} \Delta r (\mathbf{J}^{-1} \Delta r)^T\} = \mathbf{J}^{-1} \text{Cov}(r) \mathbf{J}^{-T} = \mathbf{J}^{-1} \Sigma_r \mathbf{J}^{-T} \quad (\text{Ec. 20})$$

En el caso de que todas las medidas de distancias o diferencias de distancias sean independientes y tengan similares errores, la covarianza de la matriz de distancia 'r', devuelve como resultado una matriz identidad cuya diagonal es la varianza de ruido que se introduce en el sistema, por tanto, la ecuación anterior se reduce de la siguiente manera:

$$\text{Cov}(\mathbf{X}) = \sigma_0^2 (\mathbf{J}^T \mathbf{J})^{-1} \quad (\text{Ec. 21})$$

De esta matriz de covarianza se pueden obtener las varianzas de las coordenadas de posición que se necesitan para el cálculo del PDOP, por tanto:

$$\begin{aligned}
 PDOP * \sigma_0 &= \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} = \sqrt{\text{trace}(\text{Cov}_X)} = \sqrt{\text{trace}(\sigma_0^2 * (\mathbf{J}^T * \mathbf{J})^{-1})} \\
 &= \sigma_0 * \sqrt{\text{trace}((\mathbf{J}^T * \mathbf{J})^{-1})} \\
 PDOP &= \sqrt{\text{trace}((\mathbf{J}^T * \mathbf{J})^{-1})}
 \end{aligned}$$

(Ec. 22)

Como se puede observar, este cálculo de PDOP mediante el empleo de la función inversa es independiente del método utilizado para el cálculo de la matriz **A**. Se puede calcular esta matriz mediante el uso de la trilateración esférica e hiperbólica usando ambos métodos de resolución, por linealización o Gauss-Newton.

3.3 PDOP mediante el cálculo de la altura de tetraedros

Como un recurso geométrico adicional para obtener una medida fiable de PDOP mediante otros métodos, se intenta calcular esta medida a través de la teoría del uso del volumen del tetraedro formado por 4 balizas y el punto en el que se sitúa el móvil, trasladando la posición de las balizas a una esfera unitaria con centro en la posición del móvil y mismos vectores directores hacia las balizas. Con ello se consigue calcular la medida del PDOP mediante la siguiente manera:

$$PDOP = \sqrt{\frac{1}{h_A^2} + \frac{1}{h_B^2} + \frac{1}{h_C^2} + \frac{1}{h_D^2}}$$

(Ec. 23) [2]

Donde se necesita calcular previamente las distintas alturas del tetraedro formado por los cuatro puntos en la esfera unitaria. Las variables h_A^2 , h_B^2 , h_C^2 y h_D^2 hacen referencia a las cuatro alturas de dicho tetraedro.

A través de esta teoría se corroborará que se puede hacer buen uso de esta idea, pero atendiendo a distintas limitaciones. La principal de ellas es que el tetraedro se tiene que formar en un principio con los puntos que delimitan una esfera unitaria en las direcciones que van del usuario a cada una de las balizas empleadas.

Las ideas que se acaban de comentar se resumen con la siguiente ilustración:

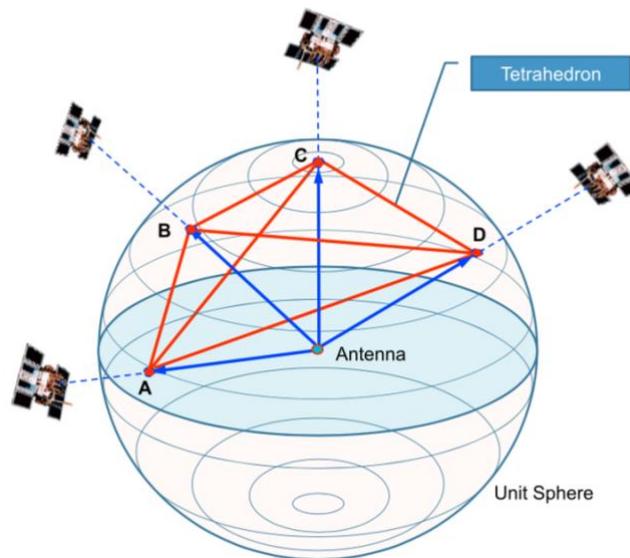


Ilustración 6. Ejemplo de cálculo de la altura de tetraedros con la esfera de radio unidad [2].

Suponiendo que los satélites de la ilustración sean las balizas del sistema de referencia que se va a emplear, y la antena el punto “real” del que se pretende calcular su posición, se dibuja una esfera de radio unidad alrededor del móvil, y se calculan los nuevos puntos que representarán las nuevas balizas de distancia unidad. Esos puntos nuevos son los que se representan en la ilustración con las letras ‘A’, ‘B’, ‘C’ y ‘D’. Con esas nuevas balizas se construye un tetraedro (en rojo en la ilustración) y se calculan las cuatro alturas del tetraedro (desde cada vértice al lado opuesto) para posteriormente calcular el PDOP.

Hay que tener en cuenta que cada vez que se cambie de punto “real”, hay que volver a calcular el nuevo sistema para que se ajuste a esas condiciones mencionadas.

Por otro lado, cabe destacar que, debido a que para esta teoría se necesitan formar tetraedros, se necesitan 4 balizas para conseguir formar ese volumen proyectándolo sobre la esfera unitaria desde el punto del que se va a calcular la posición. Como se tiene un sistema de 5 balizas, para cada punto a estimar su posición se tendrán cinco combinaciones diferentes de balizas, por lo que se tendrán cinco tetraedros diferentes a los que se habrá de calcular las alturas para el uso de la Ecuación 23 (al final se promediarán estos resultados). Una vez calculado el PDOP para una posición, se procederá a mostrar un mapa de PDOP a través de las diferentes medidas de precisión que se irán calculando a través de las distintas posiciones donde se quiera calcular esa medida.

3.3.1 Bideterminante Cayley-Menger

En este apartado, se propone la idea de la aplicación de los Bideterminantes de Cayley-Menger para volver a calcular la medida de PDOP a través de las alturas de los distintos tetraedros como se ha hecho en el apartado anterior. Ahora, se propone el uso de los bideterminantes como sustitución de esa función empleada, evitando así, el uso de funciones trigonométricas, ya que los mismos utilizan sólo distancias entre puntos. Se va a describir en qué consisten esos bideterminantes, y cómo se puede calcular lo comentado.

Se define el bideterminante de Cayley-Menger de dos conjuntos de puntos \mathbf{p} y \mathbf{q} (genéricamente con n puntos cada conjunto) como:

$$D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{q}_1, \dots, \mathbf{q}_n) = 2 \left(\frac{-1}{2} \right)^n \begin{vmatrix} 0 & 1 & 1 & \dots & 1 \\ 1 & D(\mathbf{p}_1, \mathbf{q}_1) & D(\mathbf{p}_1, \mathbf{q}_2) & \dots & D(\mathbf{p}_1, \mathbf{q}_n) \\ 1 & D(\mathbf{p}_2, \mathbf{q}_1) & D(\mathbf{p}_2, \mathbf{q}_2) & \dots & D(\mathbf{p}_2, \mathbf{q}_n) \\ \dots & \dots & \dots & \dots & \dots \\ 1 & D(\mathbf{p}_n, \mathbf{q}_1) & D(\mathbf{p}_n, \mathbf{q}_2) & \dots & D(\mathbf{p}_n, \mathbf{q}_n) \end{vmatrix} \quad (\text{Ec. 24})$$

Donde $D(\mathbf{p}_n, \mathbf{q}_n)$ se define como el cuadrado de la distancia entre los puntos indicados, esto es, $D(\mathbf{p}_n, \mathbf{q}_n) = d(\mathbf{p}_n, \mathbf{q}_n)^2$, donde $d()$, representa la distancia entre los puntos.

Cuando las secuencias de los puntos \mathbf{p} y \mathbf{q} son las mismas, se puede representar el bideterminante de la siguiente manera:

$$D(\mathbf{p}_1, \dots, \mathbf{p}_n; \mathbf{p}_1, \dots, \mathbf{p}_n) = D(\mathbf{p}_1, \dots, \mathbf{p}_n) \quad (\text{Ec. 25})$$

Cuando ocurre esto, al bideterminante se le denomina Determinante de Cayley-Menger. Este determinante posee múltiples propiedades geométricas que se deben tener en cuenta para desarrollar el cálculo del mismo, y sobre todo, para entender cómo se va a aplicar estos determinantes a este caso práctico.

Cuando se tienen 2 puntos, que es el caso de $n=2$, se representa el determinante como la distancia al cuadrado de esos dos puntos debido a que éstos definen una recta.

$$D(\mathbf{p}_1, \mathbf{p}_2) = d(\mathbf{p}_1, \mathbf{p}_2)^2 \quad (\text{Ec. 26})$$

Como se puede comprobar, esto concuerda con la propia definición de los elementos de los bideterminantes.

Para el caso de $n=3$, se puede definir un área con esos tres puntos que se tienen (un triángulo). Esa área se puede representar de la siguiente manera:

$$D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3) = 4A^2 = \|(\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)\|^2 \quad (\text{Ec. 27})$$

La ecuación anterior representa el cuadrado del doble del área del triángulo definido por los tres puntos que se tienen.

Para el caso de $n=4$, se puede definir un volumen de un tetraedro formado a partir de los 4 puntos que se tienen. Ese volumen se puede representar como:

$$D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4) = 36V^2 \quad (\text{Ec. 28})$$

La ecuación anterior define el volumen construido a partir de cuatro puntos como el cuadrado de seis veces el volumen del tetraedro.

Con estas propiedades de los bideterminantes ya conocidas, se puede empezar a plantear el cálculo del PDOP mediante la aplicación de esta técnica. La siguiente ilustración, muestra un esquema de lo que se pretende calcular:

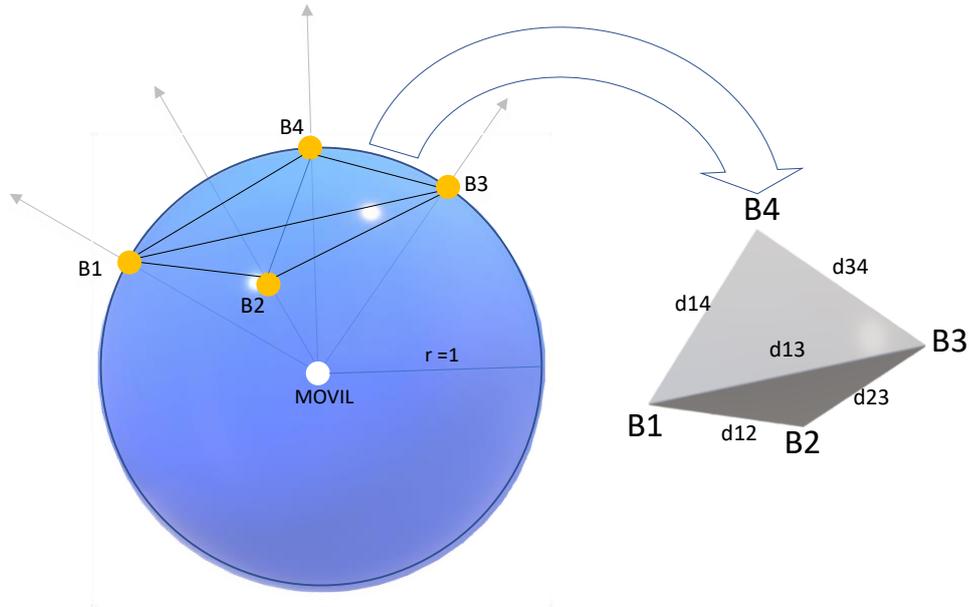


Ilustración 7. Esquema básico del problema de la trilateración [4].

Como es visible en la ilustración, se forma un tetraedro con cuatro balizas, a partir del cual se calcularán las cuatro alturas que tiene el volumen formado.

Se sabe que el volumen de un tetraedro se define matemáticamente como:

$$V_T = \frac{1}{3} A_b h$$

(Ec. 29)

Donde A_b representa el área de la base de la figura y h representa la altura de la figura formada.

Teniendo en cuenta la ecuación anterior y las propiedades geométricas comentadas anteriormente, especialmente las que tienen que ver con la Ecuación 27 y la Ecuación 28, se puede definir la altura del tetraedro (formado por los puntos $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4)$ sobre la base formada por $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$) como:

$$h = \frac{\sqrt{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4)}}{\sqrt{D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)}} \quad (\text{Ec. 30})$$

Por tanto, y viendo la propia definición de lo que son los bideterminantes, se puede definir cada determinante presente en la Ecuación 30, y asignarlo a los puntos de la ilustración 7, como:

$$D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4) = 2 \left(\frac{-1}{2} \right)^4 \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & d_{12}^2 & d_{13}^2 & d_{14}^2 \\ 1 & d_{12}^2 & 0 & d_{23}^2 & d_{24}^2 \\ 1 & d_{13}^2 & d_{23}^2 & 0 & d_{34}^2 \\ 1 & d_{14}^2 & d_{24}^2 & d_{34}^2 & 0 \end{vmatrix} = D(\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3, \mathbf{B}_4) \quad (\text{Ec. 31})$$

$$D(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3) = 2 \left(\frac{-1}{2} \right)^3 \begin{vmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & d_{12}^2 & d_{13}^2 \\ 1 & d_{12}^2 & 0 & d_{23}^2 \\ 1 & d_{13}^2 & d_{23}^2 & 0 \end{vmatrix} = D(\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3) \quad (\text{Ec. 32})$$

En estos determinantes se define cada elemento de la matriz como la distancia al cuadrado entre cada uno de los puntos indicados. Se recuerda, que los puntos \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 y \mathbf{p}_4 representan las balizas con las que se forma el tetraedro. Cabe destacar también que, las diagonales de las matrices son cero porque representan la distancia de un punto a sí mismo.

También cabe recordar que en este método de uso de alturas se había puesto una restricción, comentada previamente, que consiste en que, para calcular dicha altura, se tenían que construir esferas de radio unidad con centro el punto a posicionar, por lo que las balizas tendrían que estar en esa misma esfera. Esto consigue que se tengan que recalculer los puntos que se utilizan como balizas en la proyección de la recta con el punto hasta conseguir distancia unidad (recordar Ilustración 6 y 7).

En estos dos últimos métodos donde se trabaja con el cálculo de la altura, se ha comentado que se trabaja con las proyecciones de las balizas en los puntos que cumplen con la distancia de radio unidad desde el móvil. Para recordar, la situación de las nuevas balizas se calculará como:

$$\begin{aligned}\vec{B}_{nueva} &= \vec{p}_{movil} + \vec{d}_{unidad} \\ \vec{d}_{unidad} &= \frac{\vec{p}_{movil} - \vec{B}_{original}}{|\vec{p}_{movil} - \vec{B}_{original}|}\end{aligned}$$

(Ec. 33)

Donde \vec{B}_{nueva} representa el nuevo punto de la baliza que cumple con la distancia de radio unidad desde el móvil, \vec{p}_{movil} indica la posición del móvil desde el centro de coordenadas del sistema, y $\vec{B}_{original}$ indica la posición de la baliza desde el centro de coordenadas del sistema.

La siguiente ilustración muestra un recordatorio de lo que se acaba de comentar:

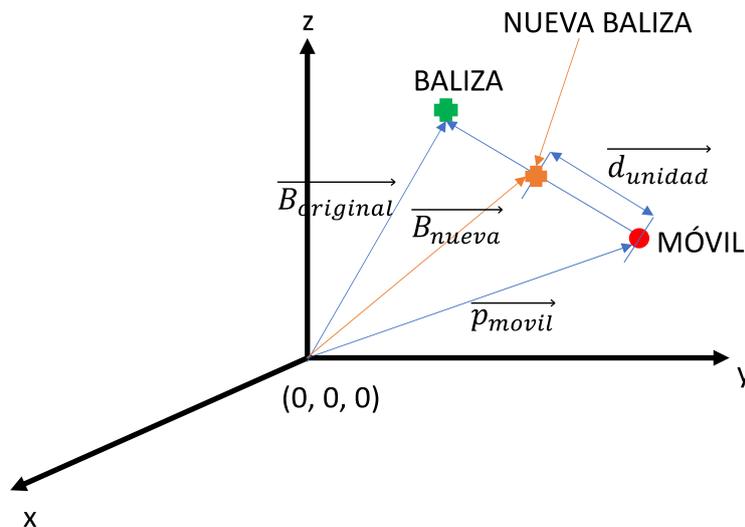


Ilustración 8. Cálculo de nuevas balizas con distancia de radio unidad desde el móvil.

4 ESQUEMA PARA SIMULACIÓN 3LPS

Dentro de los métodos que se han descrito anteriormente, se va a destacar el método Gauss-Newton, tanto esférico como hiperbólico, para la realización de las siguientes pruebas de simulación, dado que es uno de los más utilizados en posicionamiento.

Para la realización de estas pruebas, se han realizado varios esquemas de simulación para probar los distintos modelos a analizar.

Antes de empezar a comentar esos modelos, se va a proceder a la explicación del entorno que se ha creado para la simulación de los distintos escenarios. Las pruebas se han centrado en una **habitación de 5 metros por 5 metros por 5 metros**. Se plantea un sistema de referencia que se compone de un conjunto de 5 balizas con el siguiente esquema que se presenta a continuación:

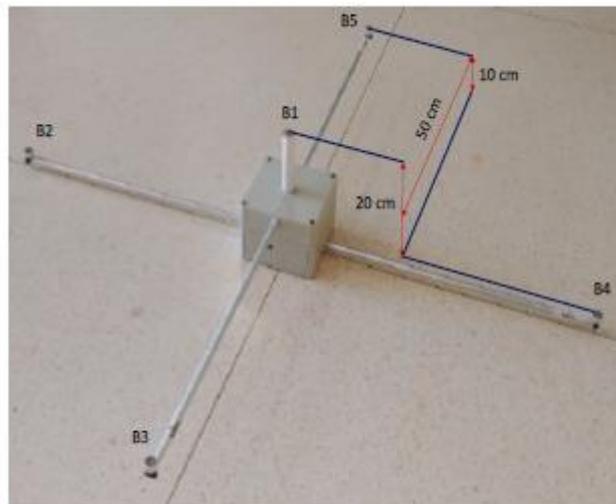


Ilustración 9. Sistema de balizas a utilizar [1].

Cuando se trata de sistemas de posicionamiento, es muy importante conocer la precisión con la que se obtiene la posición estimada. Esto depende tanto de la calidad de las medidas de distancias desde las balizas al objetivo, como de la posición y geometría del sistema de balizas que se utiliza como referencia. Es por ello por lo que se usa la medida de PDOP y se ha determinado el sistema de referencia con la geometría que se puede visualizar en la ilustración anterior. Con esa geometría, se evita que las balizas sean coplanares, y así, se evitan también las singularidades que se pueden presentar al intentar obtener el posicionamiento con los algoritmos comentados en los apartados anteriores.

Para comenzar, se va a posicionar el sistema de balizas en el plano ' $z=5$ ' para empezar a obtener la posición de múltiples puntos que se encuentran situados a lo largo de la habitación planteada como entorno de simulación.

La siguiente ilustración 10 muestra la situación que se acaba de describir. En ella se puede observar el conjunto de balizas que se ha descrito en color azul, y se muestran tres planos diferentes donde se van a obtener distintas posiciones del móvil (color negro, verde y rojo). Dentro de cada plano se pueden distinguir 9 puntos diferentes donde se intentará posicionar el objetivo. En dicha figura se ve también dónde se ha considerado el origen de coordenadas (0, 0, 0).

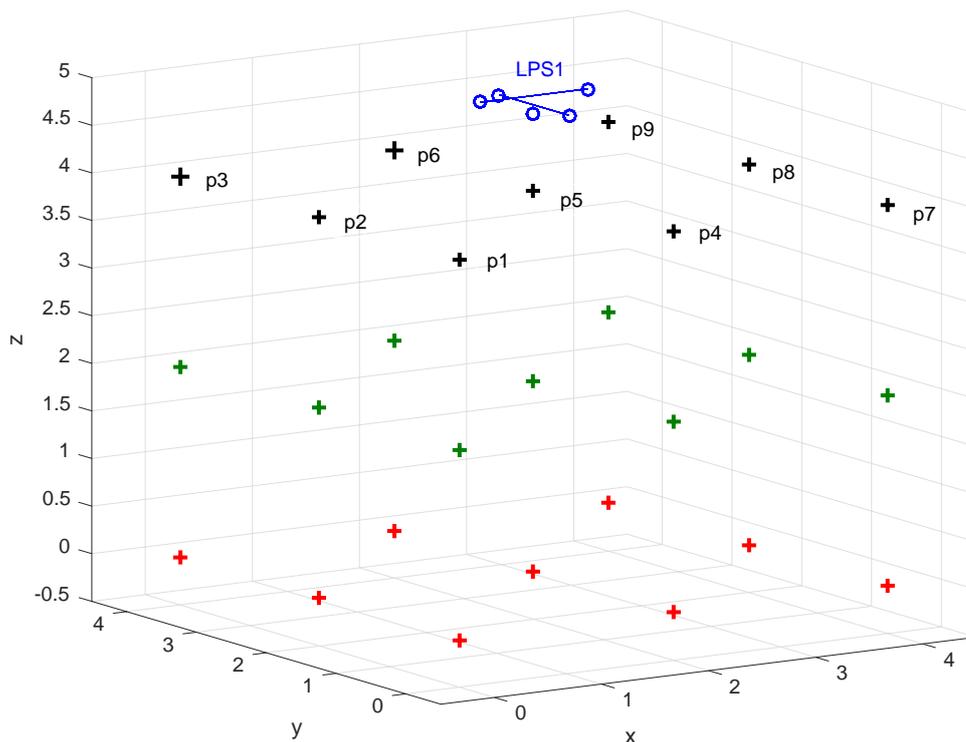


Ilustración 10. Esquema del entorno de simulación con un conjunto de balizas.

Como se puede apreciar de la ilustración anterior, el sistema de balizas se encuentra en el centro del plano ' $z=5$ ' con la siguiente situación de cada baliza que respeta las medidas mostradas anteriormente:

```

%Posición de las balizas
%      B1  B2  B3  B4  B5
bal = [ 2    1.5  2    2.5  2;           %Coordenadas X de las balizas
        2    2    1.5  2    2.5;       %Coordenadas Y de las balizas
        4.8  5    4.9  5    4.9];     %Coordenadas Z de las balizas

```

Ilustración 11. Posición de las balizas.

Una vez comprobado que se es capaz de posicionar el móvil correctamente con una medida de PDOP razonable, se van a probar diversos modos de simulación para comprobar si se puede mejorar la precisión del posicionamiento a partir del uso de varios sistemas de balizamiento y de distintas maneras de combinación de ese nuevo conjunto de datos.

Para ello, se va a colocar un sistema de balizas (LPS de cinco balizas) en cada plano ' $x=5$ ', ' $y=5$ ' y ' $z=5$ ' desde los cuales se van a realizar las distintas medidas de manera simultánea o de manera independiente. Esto es lo que se representa en la siguiente ilustración:

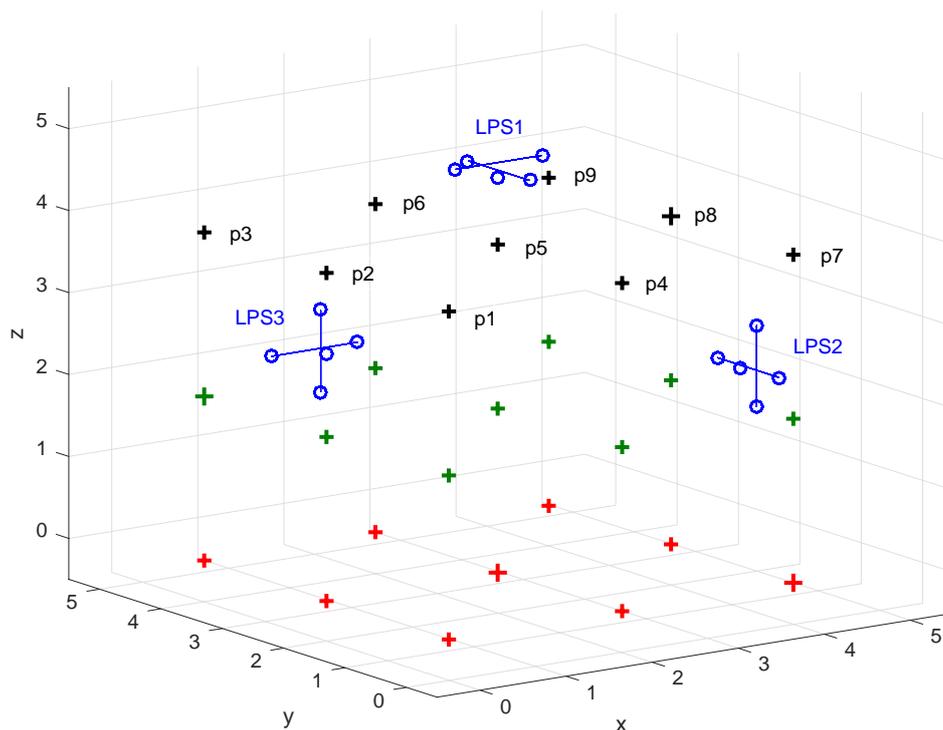


Ilustración 12. Esquema del entorno de simulación con tres conjuntos de balizas.

A continuación, se va a comentar cada modelo de simulación para ver las diferencias de cada uno de ellos.

4.1 Modo individual/independiente

Aplicando este modo de simulación, se quieren recoger los resultados de posicionamiento de un punto determinado en el entorno de los 3 sistemas de balizas de manera independiente para, posteriormente, unificarlos con diversos métodos de fusión de los que se hablarán en el siguiente apartado.

Como se hace una medida independiente con cada sistema de balizas (cada LPS), no hay necesidad de repetir el mismo proceso 3 veces. Obtenidos los resultados de posicionamiento de los blancos en el entorno a través de un solo conjunto de balizas, esa matriz de resultados se puede rotar y ordenar de manera que se convierta el sistema de referencias del que se han recogido las primeras medidas ($z=5$), en el nuevo sistema en el que se va a continuar recogiendo medidas ($x=5$ o $y=5$). Con eso se tendrían los datos de cada punto del entorno recogidos de los 3 sistemas de balizas diferentes. Únicamente quedaría elegir un método de fusión de datos para que las medidas de un único punto puedan converger en una medida que sería más precisa. El método elegido para esa fusión de datos se ha desarrollado de acuerdo a un algoritmo que se comentará posteriormente.

La manera de rotar los datos consiste únicamente en cambiar el eje de referencia de éstos, por lo que la rotación quedaría de la siguiente manera:

LPS1 ($z=5$):	LPS2 ($x=5$):	LPS3 ($y=5$):
$X_{z=5} = X_{\text{real}}$	$X_{x=5} = Z_{\text{real}}$	$X_{y=5} = X_{\text{real}}$
$Y_{z=5} = Y_{\text{real}}$	$Y_{x=5} = Y_{\text{real}}$	$Y_{y=5} = Z_{\text{real}}$
$Z_{z=5} = Z_{\text{real}}$	$Z_{x=5} = X_{\text{real}}$	$Z_{y=5} = Y_{\text{real}}$

Ilustración 13. Rotación de los ejes para posicionar los distintos sistemas de balizas.

Donde los datos ' x_{real} ', ' y_{real} ', ' z_{real} ' hacen referencia a los ejes naturales del sistema de coordenadas de centro (0,0,0).

4.2 Modo simultáneo

Con este método de simulación, se quieren recoger a la vez las medidas realizadas a través de los 3 sistemas de balizas al mismo tiempo para obtener un único resultado de posicionamiento de un cierto punto (se aplicaría el algoritmo de posicionamiento considerando todas las medidas de distancias a las balizas de los tres LPSs). Con ello no haría falta desarrollar a posteriori un algoritmo de fusión de datos debido a que con este modo se estima una posición única con todas las medidas realizadas.

Al utilizar los 3 sistemas de balizamiento en el mismo instante de tiempo, se necesita resolver un sistema de 15 ecuaciones, resultantes de las 15 balizas, que se utilizan para la obtención de las medidas desde el móvil a cada baliza. Con todas esas medidas, y aplicando uno de los algoritmos de posicionamiento comentados, se estima de manera más precisa una posición del objeto móvil.

Para el desarrollo de este modo se emplea lo expuesto en la Ilustración 12 donde se visualizan los 3 sistemas de balizas en sus posiciones de referencia.

5 ALGORITMO DE FUSIÓN

Este algoritmo se aplica para obtener una fusión de los datos obtenidos a través de la realización de medidas provenientes de 3 sistemas de balizas de manera independiente (modo independiente anteriormente comentado). Como resultado de este modo, se obtenía el posicionamiento en un punto del entorno mediante un sistema básico de 5 balizas, que posteriormente, se rotaba, para calcular en el mismo punto el posicionamiento debido a la colocación del sistema de balizas en otros planos.

Para esa rotación de datos que se comenta, hay que recordar lo que se refleja en la Ilustración 13, que es simplemente el cambio entre las variables para los nuevos sistemas de coordenadas.

La manera de fusionar los diferentes datos, obtenidos de los 3 sistemas de referencia colocados en los distintos planos, es mediante la combinación de las varianzas resultantes de cada punto estimado de posicionamiento de cada sistema de baliza independiente.

Esta manera de fusionar los datos se denomina método MLE, que es el método de estimación por máxima verosimilitud, debido a que se pondera cada dato según su varianza. Eso se consigue, para el caso de tres conjuntos de datos, mediante las siguientes fórmulas:

$$X_{MLE} = \frac{\sigma_{1x}^{-2}x_1 + \sigma_{2x}^{-2}x_2 + \sigma_{3x}^{-2}x_3}{\sigma_{1x}^{-2} + \sigma_{2x}^{-2} + \sigma_{3x}^{-2}}$$

$$Y_{MLE} = \frac{\sigma_{1y}^{-2}y_1 + \sigma_{2y}^{-2}y_2 + \sigma_{3y}^{-2}y_3}{\sigma_{1y}^{-2} + \sigma_{2y}^{-2} + \sigma_{3y}^{-2}}$$

$$Z_{MLE} = \frac{\sigma_{1z}^{-2}z_1 + \sigma_{2z}^{-2}z_2 + \sigma_{3z}^{-2}z_3}{\sigma_{1z}^{-2} + \sigma_{2z}^{-2} + \sigma_{3z}^{-2}}$$

(Ec. 34) [1]

Donde se tiene en cuenta la posición de cada punto calculado de cada sistema de balizas diferente y la desviación típica de cada coordenada.

Para comprenderlo mejor, se va a definir cada variable utilizada en el algoritmo. El punto 1 se contempla como (x_1, y_1, z_1) , el punto 2 se contempla como (x_2, y_2, z_2) y el punto 3 se contempla como (x_3, y_3, z_3) . Cada punto comentado es una posición estimada de un mismo punto "real", sacada de un algoritmo de posicionamiento como puede ser el de Gauss-Newton, que se obtiene a partir de un sistema de balizas diferente. Por ejemplo, el punto 1 se ha obtenido del sistema de balizas de 'z=5', el punto 2 se ha obtenido del sistema de balizas situado en el plano 'x=5' y el punto 3 se obtiene de las balizas situadas en el plano 'y=5'.

A partir de cada punto obtenido, se puede obtener, mediante múltiples simulaciones previas, la matriz de covarianza para poder calcular la variación conjunta de las tres variables de posicionamiento que se necesitan. A través de esa covarianza se obtienen las varianzas de cada coordenada de cada uno de los puntos que se quieren fusionar; quedando, como ejemplo, las varianzas de cada coordenada del punto 1 de la siguiente manera $(\sigma_{1x}^2, \sigma_{1y}^2, \sigma_{1z}^2)$. Con eso se tiene toda la información necesaria para aplicar el algoritmo de fusión según las fórmulas anteriores expuestas, obteniendo así el posicionamiento final del usuario.

Hay que recordar que, al fusionar los datos que se han obtenido de manera independiente de cada sistema, la precisión de estos será menos certera que la precisión que se podría obtener calculando una posición a través de varias mediciones de manera simultánea con distintos sistemas.

También hay que considerar que cuantas más medidas se puedan obtener para el sistema, mejores resultados de posición se podrían obtener debido a que se tendrán más mediciones que considerar para esa fusión de datos.

6 RESULTADOS DE SIMULACIÓN

A continuación, se van a presentar los resultados que se han obtenido de la aplicación de los algoritmos y los modos de simulación comentados anteriormente. El orden a seguir para constatar la explicación de lo comentado es el siguiente: se empezará analizando el modo independiente para dar una visión global de lo que es el algoritmo y lo que se puede esperar de él. Dentro de este modo se analizarán los resultados obtenidos a partir de un sólo sistema de balizas empleado, para posteriormente pasar a los resultados donde se han utilizado los 3 sistemas de balizas y el método de fusión empleado. Por último, se analizará el modo simultáneo para contrastar los resultados con el anterior modelo, y se pasará a representar los datos según el PDOP para ver el mejor método de obtención de resultados.

Los resultados de las siguientes simulaciones son obtenidos a través del posicionamiento Gauss-Newton debido a que se ha corroborado que es mejor método que el de posicionamiento por linealización. Dentro de este posicionamiento elegido, se verán los resultados de ambas trilateraciones, esférica e hiperbólica, comentadas en puntos anteriores.

6.1 Modo independiente

6.1.1 Trilateración esférica

En la siguiente ilustración, se puede observar que se tiene un sistema de 5 balizas situadas en el plano ' $z=5$ ' desde el cuál se realiza el posicionamiento de distintos puntos del entorno en el que el supuesto móvil se movería, a recordar, la habitación de $5m^3$. Los puntos representados en la ilustración corresponden al posicionamiento estimado que se realiza del algoritmo de Gauss-Newton. Los puntos reales que corresponderían a los puntos obtenidos están cubiertos por las nubes de diferentes colores. Para este caso, se ha calculado el posicionamiento en 3 alturas diferentes, planos ' $z=0$ ', ' $z=2$ ' y ' $z=4$ ', y dentro de un mismo plano se calculan 9 puntos diferentes a través del movimiento por el eje ' x ' e ' y ' de 2 pasos en 2 pasos.

Sobre un mismo punto a calcular se aplicará el algoritmo 50 veces para obtener varias medidas de posicionamiento con el objetivo de ver el comportamiento que tiene el algoritmo sobre el entorno establecido. Se aplicará en todas las medidas de distancia un error aleatorio de distribución gaussiana con media cero y desviación típica con valor 10^{-2} cm.

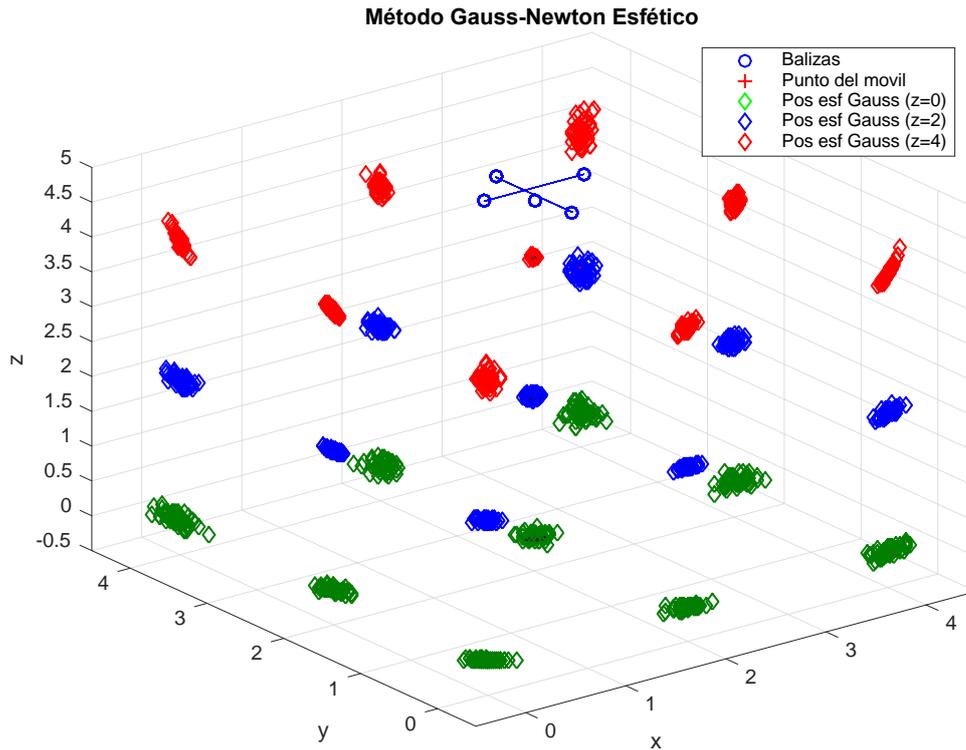


Ilustración 14. Posicionamiento en 3D con el método Gauss-Newton esférico.

Como se puede apreciar en la ilustración anterior, el posicionamiento es relativamente bueno debido a que se permite diferenciar donde se sitúan las distintas posiciones de los distintos puntos. Se puede apreciar que los puntos posicionados en color rojo corresponden al plano ' $z=4$ ', los puntos de color azul corresponden al plano ' $z=2$ ', y los puntos de color verde corresponden al plano ' $z=0$ '.

A continuación, se va a mostrar cada plano de posicionamiento de manera independiente para poder diferenciar bien cada punto.

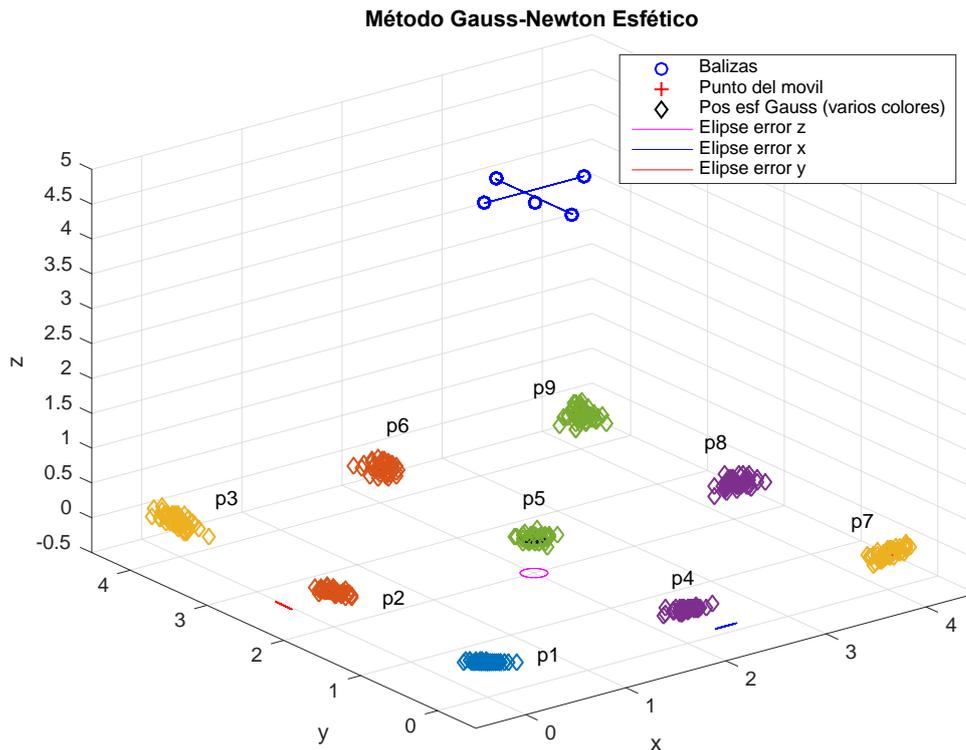


Ilustración 15. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=0$).

Se puede comprobar ahora mejor el posicionamiento que se hace de cada punto de manera independiente. De esta ilustración también se puede comentar qué clase de error se comete a la hora de posicionar. Ese error se representa a través del elipsoide de error, que es aquel que contiene el 95% de las medidas obtenidas para una misma posición. Puesto que esos elipsoides quedan cubiertos por las nubes de puntos, se representan sus proyecciones sobre cada uno de los planos de referencia para poder visualizarlo. A esas proyecciones se las denomina elipses de error.

Como se puede observar a través de esas elipses de error generadas, la mayor parte de la dispersión de los puntos se produce en el eje 'z'. Eso es debido a la situación del sistema de balizas.

En la ilustración siguiente se muestra el siguiente plano de posicionamiento. Se puede observar que se produce el mismo efecto comentado de la anterior ilustración. La dispersión de los puntos se sigue produciendo en el eje 'z' pero en menor medida. Esto se debe a que ese plano se encuentra más cerca del sistema de balizas; por lo que se puede deducir que la distancia al sistema de balizas afecta al posicionamiento.

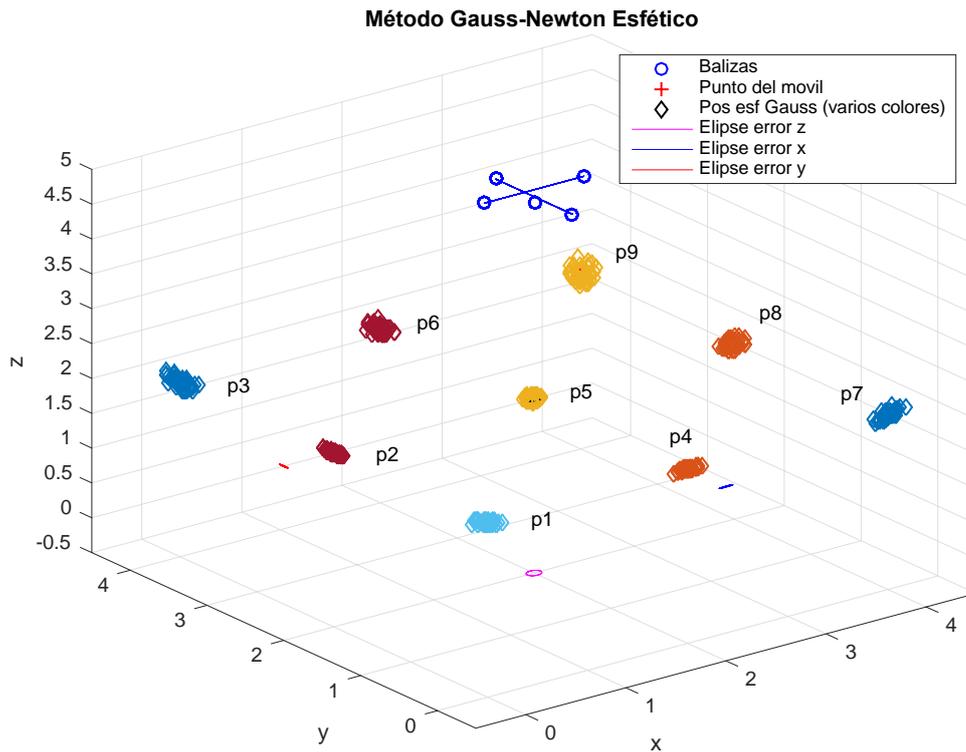


Ilustración 16. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=2$).

En la siguiente ilustración se muestra el último plano de posicionamiento donde se vuelve a corroborar que se tiene una mínima dispersión en el eje 'z'. En este caso, se puede observar que al estar muy cerca del sistema de balizas, se tienen muy buenos posicionamientos en aquellos puntos que se sitúan debajo del sistema de referencia o en las direcciones donde están situadas cada una de las balizas del sistema. Esos puntos son los denominados $p5$, $p2$, $p4$, $p6$ y $p8$. El resto de puntos sufre un poco más de dispersión que los acabados de comentar, pero nada que ver si se comparan con los mismos puntos de los otros planos.

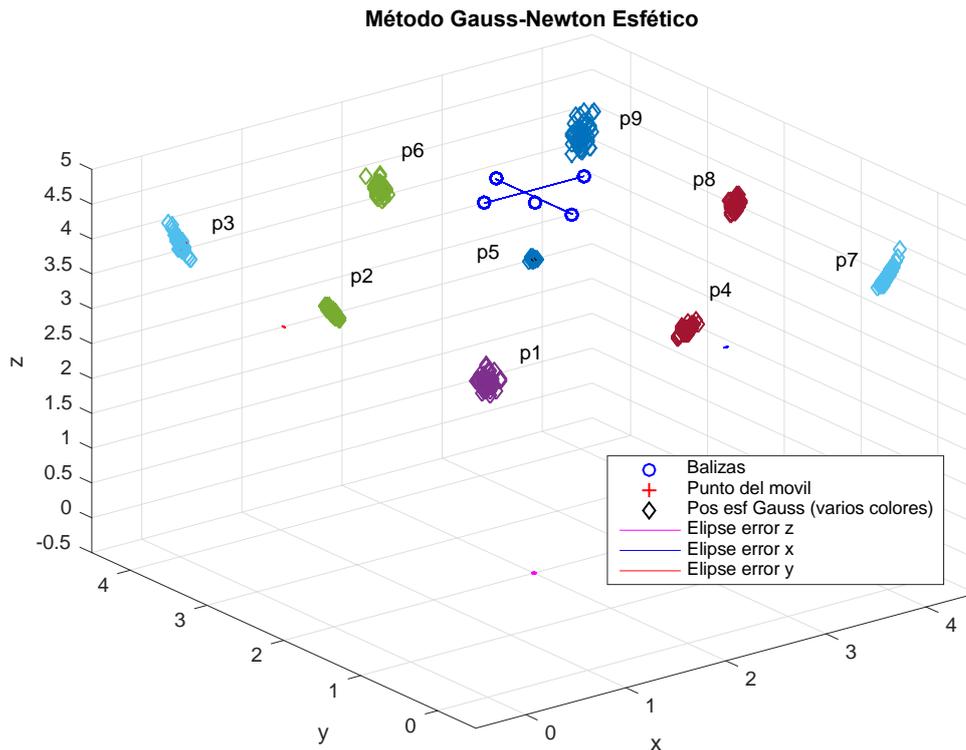


Ilustración 17. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=4$).

Se puede concluir de lo acabado de ver que, debido a la posición del sistema de balizas, se perderá precisión a la hora del cálculo de la posición estimada. Esto es debido a que según se vayan alejando los puntos a posicionar del sistema de referencia, se tendrá mayor dispersión de estos en la dirección en la que se encuentre el plano de las balizas. En las siguientes dos ilustraciones se muestra con más precisión lo acabado de comentar.

En esta primera ilustración siguiente se puede observar cómo los elipsoides de dispersión están cubiertos por las nubes de puntos. También se pueden observar las proyecciones de esos elipsoides en las caras del cubo. De la ilustración se deduce que la proyección del plano ' $z=0$ ' (circunferencia de color azul) es mayor que las proyecciones en el resto de planos; por lo que se obtiene más error en el plano que está más alejado de las balizas. En los planos ' x ' e ' y ' se puede comentar lo mismo, la circunferencia de error es mayor en el plano más alejado de las balizas.

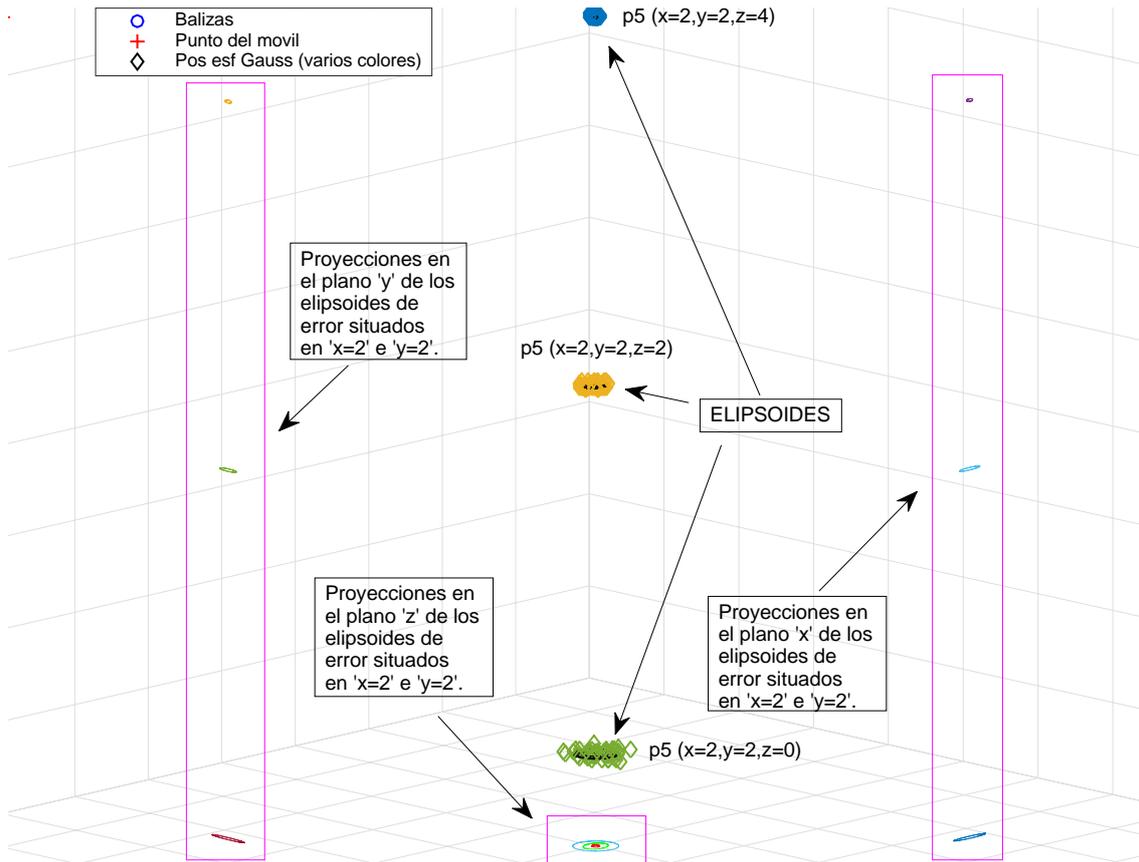


Ilustración 18. Proyecciones en todos los planos de la Ilustración 14.

En la siguiente ilustración 19 se muestra un par de elipsoides de error de dos nubes de puntos, $(x=2, y=2, z=0)$ y $(x=2, y=2, z=2)$, donde se puede ver cómo es la dispersión en 3D del posicionamiento en el punto seleccionado. Se puede comprobar que en el plano ' $z=0$ ' la nube de dispersión es mayor que la nube de dispersión correspondiente al plano ' $z=2$ '. Esto se corresponde con las proyecciones comentadas antes y con los elipsoides formados a partir de dichas nubes de puntos. También se observa mejor en la ilustración las proyecciones en el plano ' z ' de los puntos que se corresponde a la posición ' $x=2$ ' e ' $y=2$ '.

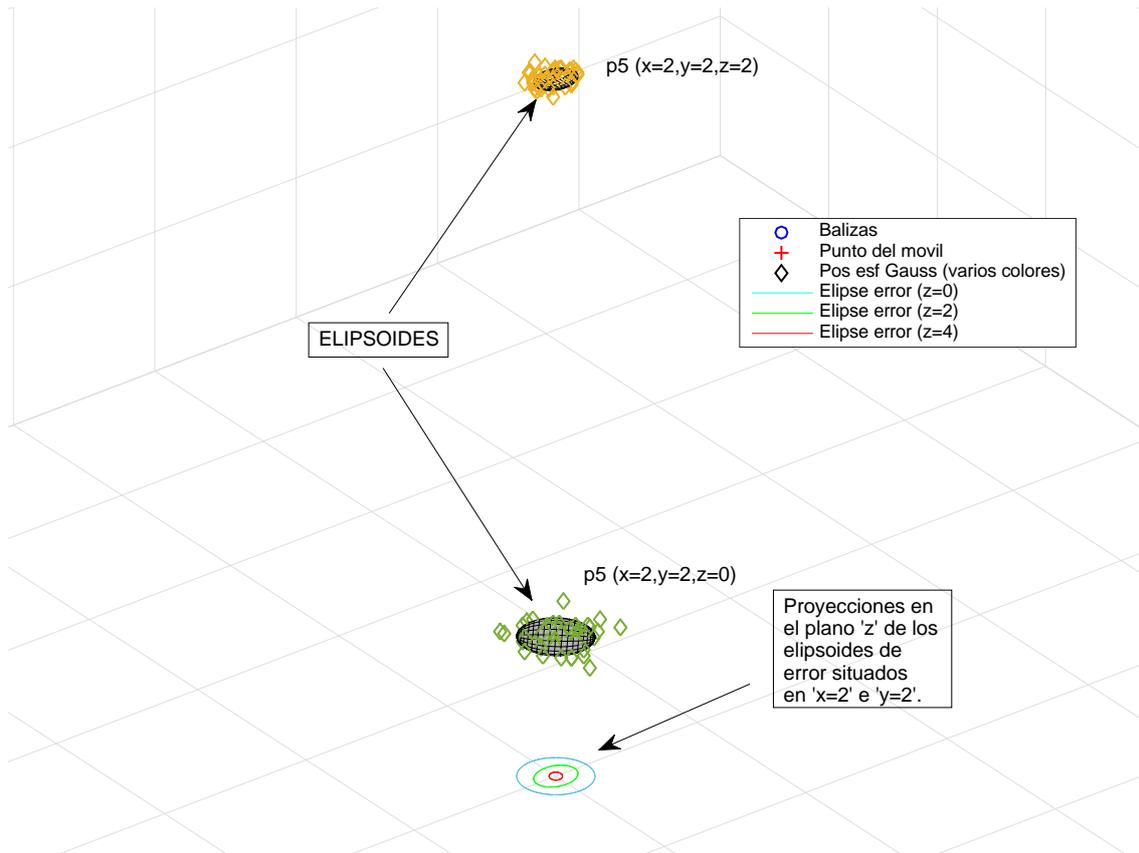


Ilustración 19. Ejemplo de elipsoides de error y proyecciones en el plano 'z=0' de la Ilustración 14.

En ambas ilustraciones aumentadas (Ilustración 18 y 19), se puede contemplar con más detalle cómo el posicionamiento es mejor en el plano más alto, 'z=4', que en el resto. De todas maneras, se comprobará en ilustraciones sucesivas que la diferencia de posición en cuanto a error cometido no es muy significativa para este caso de Gauss-Newton esférico en modo independiente.

En las próximas ilustraciones se puede observar el error cometido a la hora de posicionar en el plano 'z=2' los 9 puntos diferentes que se encuentran en ese plano; y el error cometido con la comparación del posicionamiento en los distintos planos en los que se calculan las posiciones, 'z=0', 'z=2' y 'z=4'.

Ese error se va a representar ahora a través del uso de la función acumulativa de error (CDF) que indica la probabilidad de que el error sea menor que el valor de la función en el eje de abscisas.

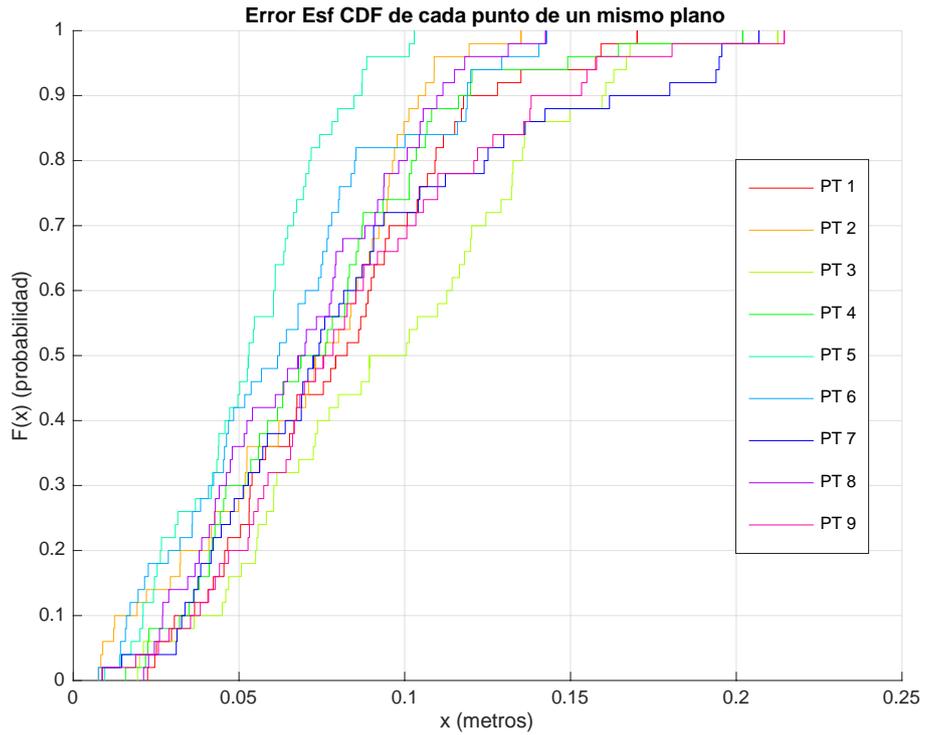


Ilustración 20. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton esférico).

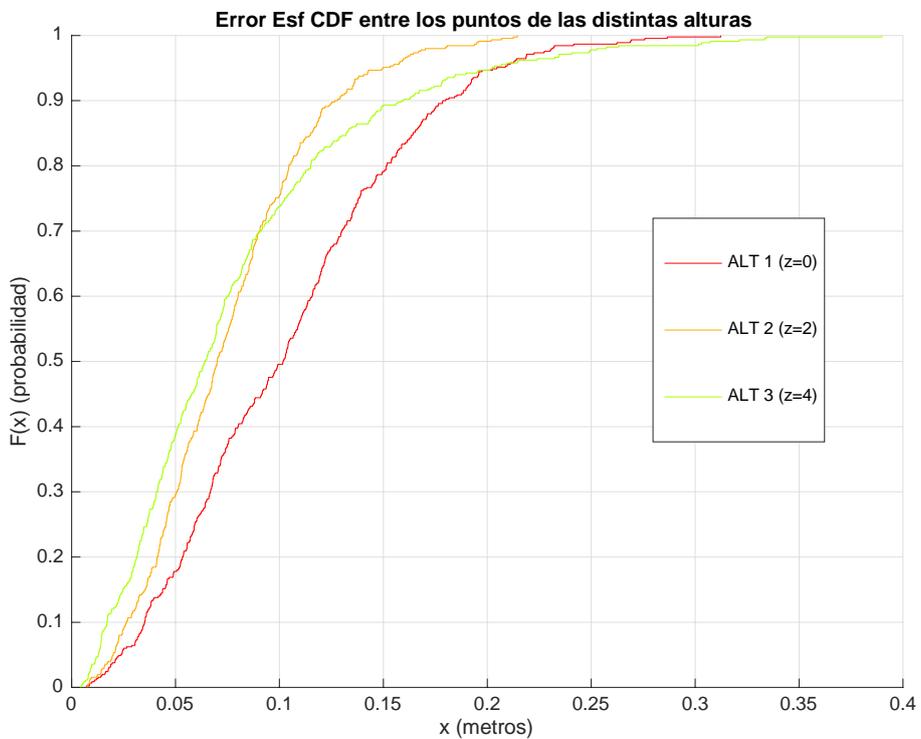


Ilustración 21. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton esférico).

Aumentando las ilustraciones anteriores, se podrá observar con más detalle lo que indican estas figuras; pero a priori se puede ver que la precisión de los puntos, tanto en un mismo plano como en los 3 planos, es muy similar; por lo que se deduce que el sistema de posición en base al sistema de referencia planteado es válido para todo el entorno.

En la siguiente ilustración, que es una ampliación de la Ilustración 20, se observa que el 90% de los puntos estimados para todas las posiciones se posicionan con un error por debajo de 18 cm. El 100% de los puntos se posicionan con una diferencia menor de 22 cm. También se puede observar que el mejor punto posicionado, el punto 5, se calcula con un error por debajo de 11 cm. Da la casualidad, que ese punto, es justo el punto que se sitúa en la misma línea que la baliza central. Si se continúa observando la ilustración, se puede concluir que los mejores puntos con un error de posición mínimo son los que se encuentran situados en las proyecciones de las balizas de los distintos planos.

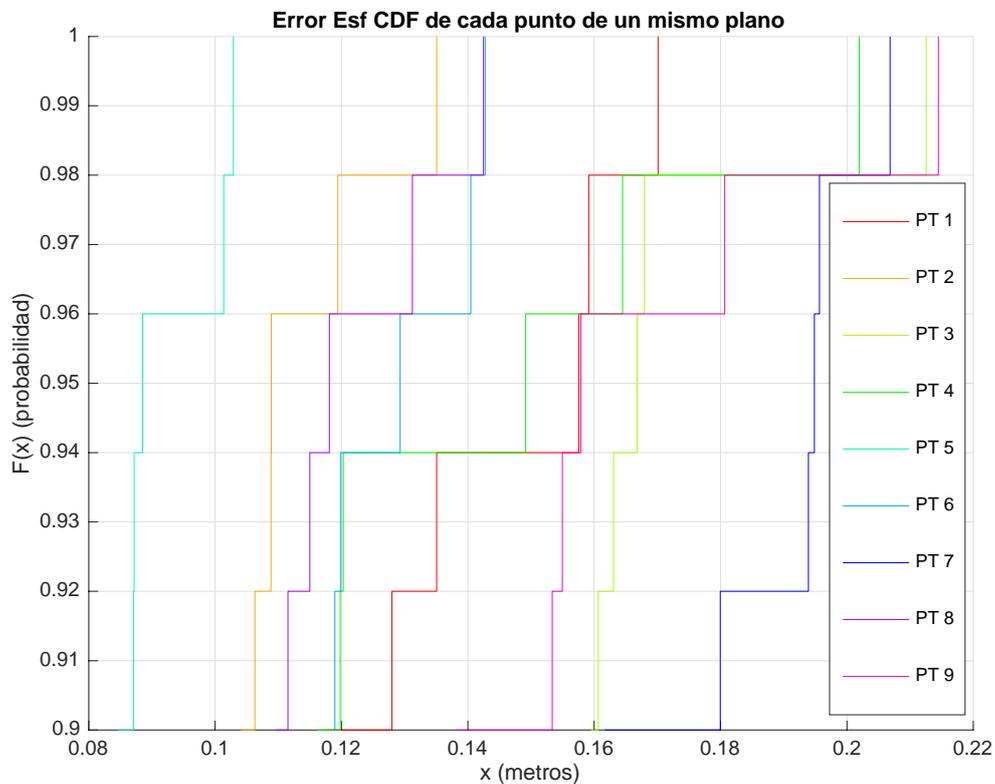


Ilustración 22. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton esférico).

En la siguiente ilustración, que es una ampliación de la Ilustración 21, se observa también que el 90% de los puntos estimados se sitúan con un error por debajo de 18 cm. El 100% de todos los puntos se sitúan con una diferencia menor de 39 cm respecto de sus posiciones originales. También se puede observar que, en el cómputo general de la ilustración, el primer plano que corresponde con ' $z=0$ ', función representada en color rojo, sufre más error que el resto de planos. Esto es debido, como se ha comentado anteriormente, que es el plano que está situado más lejos de las balizas, por lo que la precisión de las medidas de posición se ve afectada con un poco más de error. También cabe mencionar que, en el plano más cercano a las balizas, ' $z=4$ ' (función verde), hay medidas que se posicionan peor debido a una indeterminación del algoritmo, una discrepancia que puede haber en algún punto. Pero es un porcentaje mínimo como se observa en la ilustración.

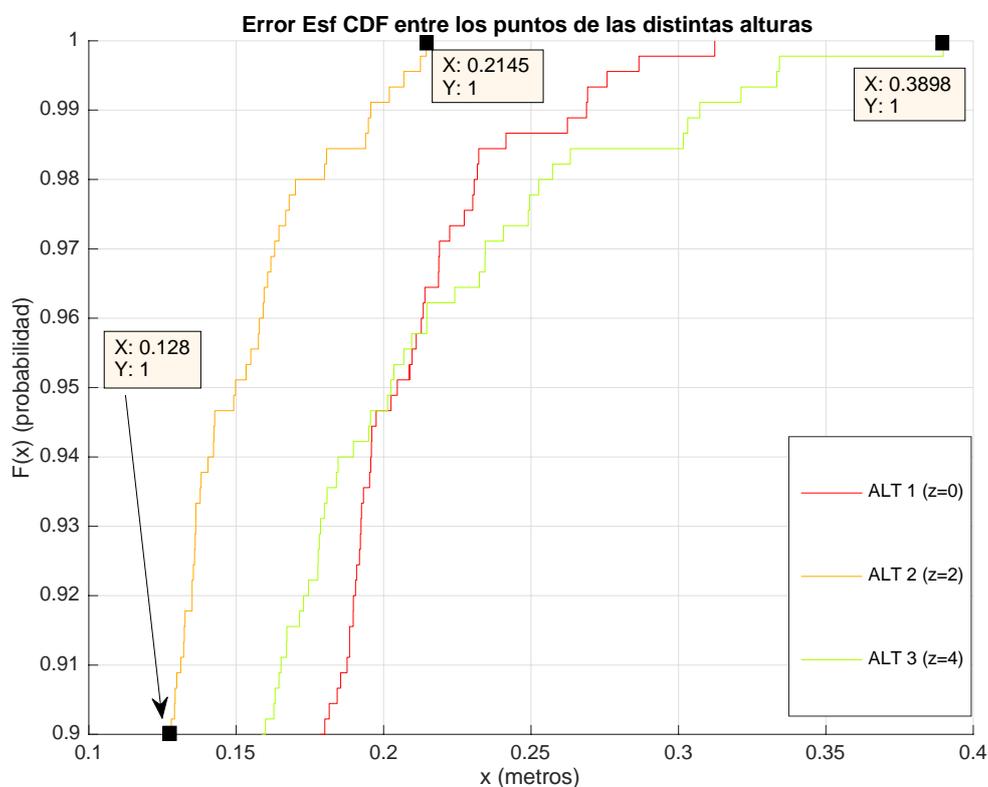


Ilustración 23. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico).

6.1.2 Trilateración hiperbólica

A continuación, se muestra el resultado del posicionamiento en los mismos puntos comentados anteriormente en los distintos planos, pero a través de la trilateración hiperbólica. Con este método, como se puede ver en las sucesivas ilustraciones, la obtención de una posición es mucho menos precisa que respecto al método anterior. Tal es esa dificultad que hay puntos en los que el algoritmo no es capaz de concretar un punto de posicionamiento definitivo. Esta conclusión será perfectamente visible gracias a los elipsoides de error y a sus proyecciones en los distintos planos.

En esta primera ilustración se muestra el posicionamiento de los distintos puntos del plano ' $z=0$ '. Cabe destacar que, con mucho error, se pueden conseguir posicionar puntos en este plano, aunque no con muy buenos resultados como se muestra. Las elipses de error muestran el error cometido en el punto central $p5$.

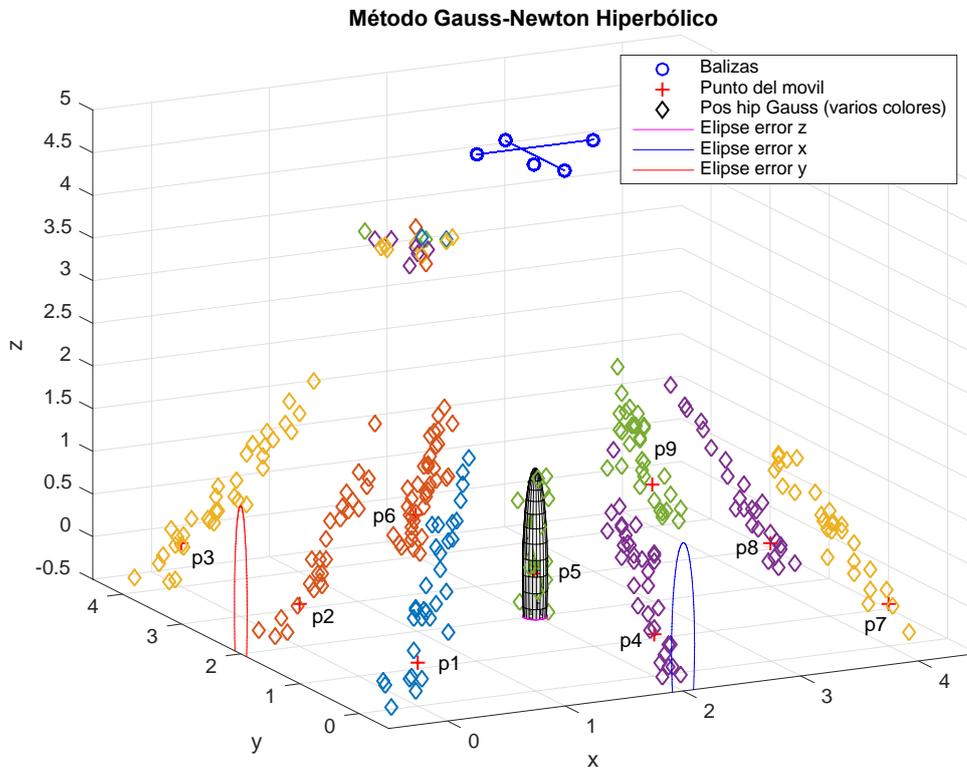


Ilustración 24. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=0$).

A continuación, se muestra el posicionamiento en el siguiente plano, ' $z=2$ '. Como se puede comprobar, ocurre lo mismo que en el plano anterior. Afortunadamente se logra un posicionamiento de todos los puntos, pero no con muy buenos resultados. Las elipses de error vuelven a mostrar el error cometido en el punto central $p5$.

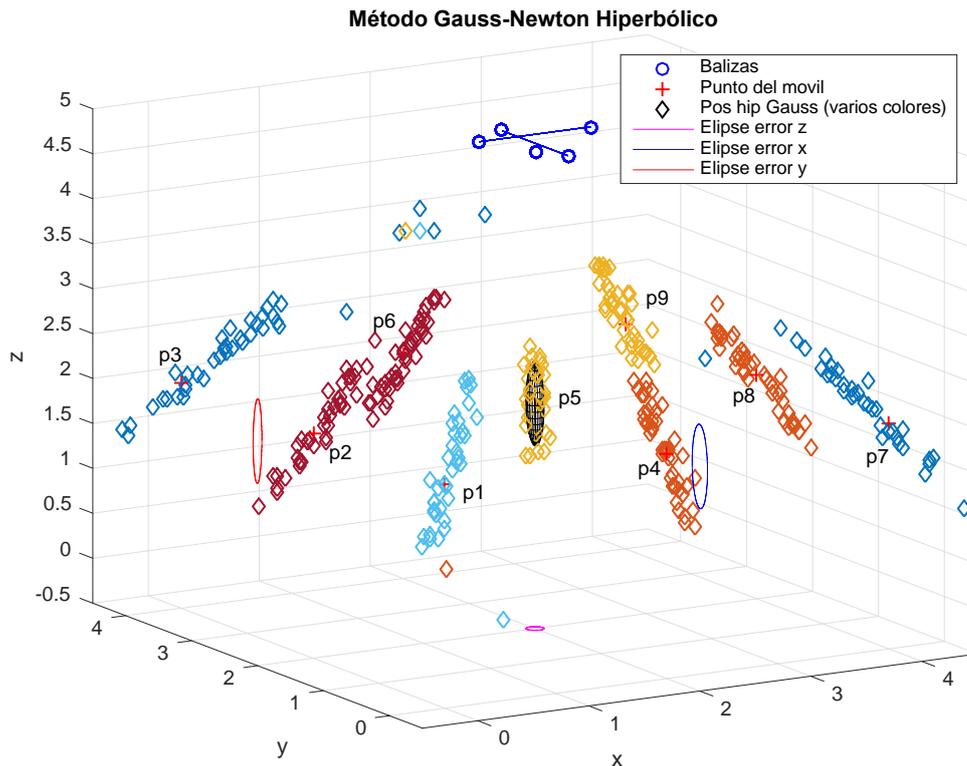


Ilustración 25. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=2$).

A continuación, se muestra el posicionamiento en el plano ' $z=4$ '. En este plano se pierde totalmente ese posicionamiento, el algoritmo no es capaz de posicionar algunos de los puntos dados. Se muestra también tanto el elipsoide de error como sus proyecciones de lo que sería el punto central puesto que no se puede saber con exactitud que nube de puntos se corresponde con ese punto debido a que están todos mezclados en una misma posición, $p1$.

La situación de todos los puntos en una misma posición tiene una explicación. Hay que remontarse a la explicación de cómo se calcula el posicionamiento por Gauss-Newton para poder entenderlo, Ilustración 5. En esa ilustración se muestra el diagrama de flujo del algoritmo donde, por medio de cálculos de incrementos de posición, se intenta minimizar el error de posición acercando una posición inicial a la posición final donde se quiere posicionar.

Para el caso hiperbólico, se tiene una condición que indica que, después de varios intentos, 100 en este caso, si todavía no se ha podido lograr una posición final donde posicionarse, se inicializa el algoritmo otra vez con una nueva posición inicial que coincide con el punto $p1$ ($x=0, y=0$). La ' z ' la determina la altura con la que se trabaja. Es en esta nueva posición donde el algoritmo si es capaz de lograr una posición, pero no de manera acertada como se puede comprobar en la ilustración siguiente.

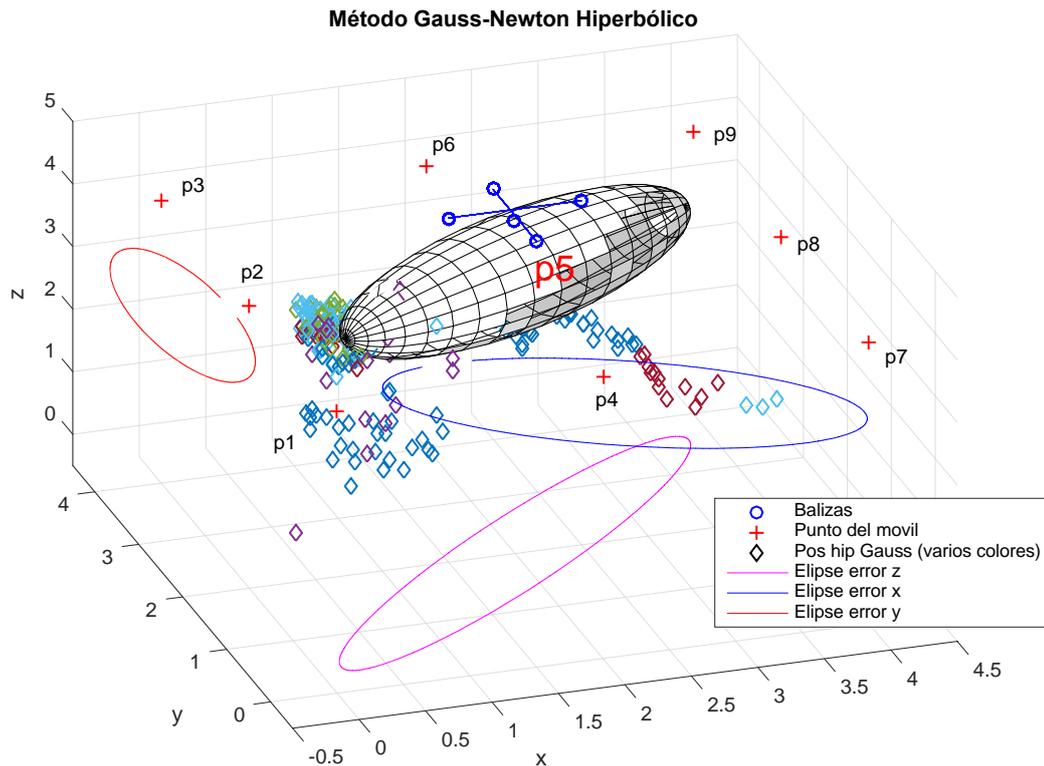


Ilustración 26. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=4$).

Unificando las ilustraciones anteriores, se puede comentar que el algoritmo es capaz de posicionar con un mínimo de decencia los puntos que están situados justo debajo de la baliza central del sistema de balizas. El resto de puntos tienen más complicado el posicionamiento debido a que el algoritmo no congenia del todo bien con la posición del sistema de balizas, por lo que se complica el cálculo de una determinada posición con un mínimo de error.

En el último plano, puede ocurrir que los puntos a posicionar se encuentren demasiado cerca de las balizas, por lo que el algoritmo no es capaz de situarse y surge ese desplazamiento de los puntos hacia la esquina donde se inicializa el algoritmo.

La siguiente ilustración, mediante la representación de los elipsoides de error y sus proyecciones, es más visible lo ya comentado. Hay una dispersión muy grande en las nubes de puntos que representan las posiciones estimadas de cada punto real a calcular. Es en el punto del plano ' $z=2$ ' donde se logra el mejor posicionamiento de todos esos puntos centrales.

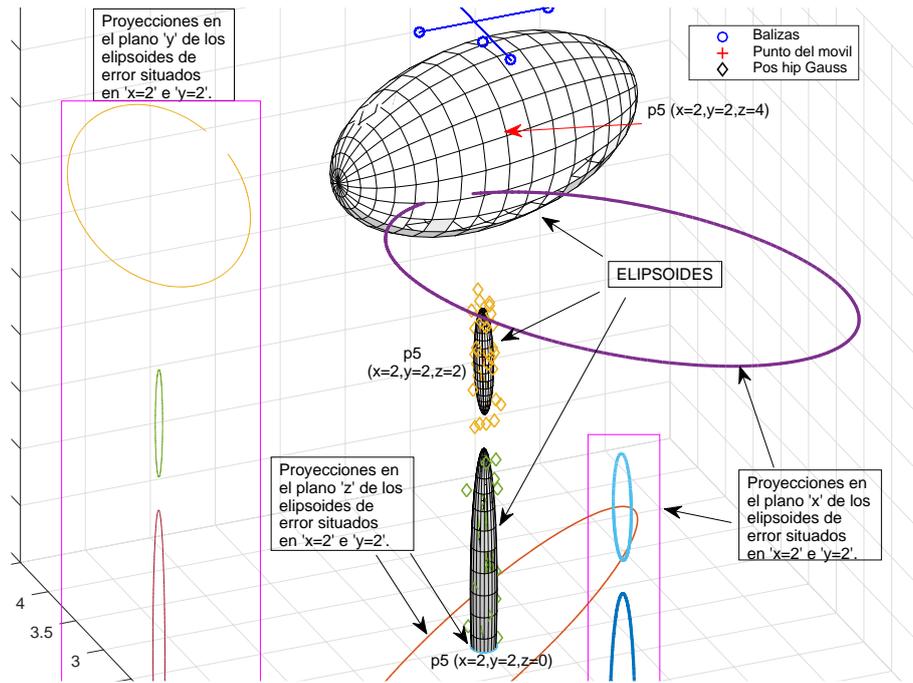


Ilustración 27. Ejemplo de elipsoides de error y sus proyecciones del método Gauss-Newton hiperbólico.

En las siguientes ilustraciones, donde se representa el error cometido a la hora de posicionar tanto en los distintos puntos de un mismo plano como en los distintos planos, se puede corroborar a simple vista que ya se puede hablar de unas magnitudes de error diferentes. En este caso, se empieza a hablar de desviaciones de posicionamiento en metros.

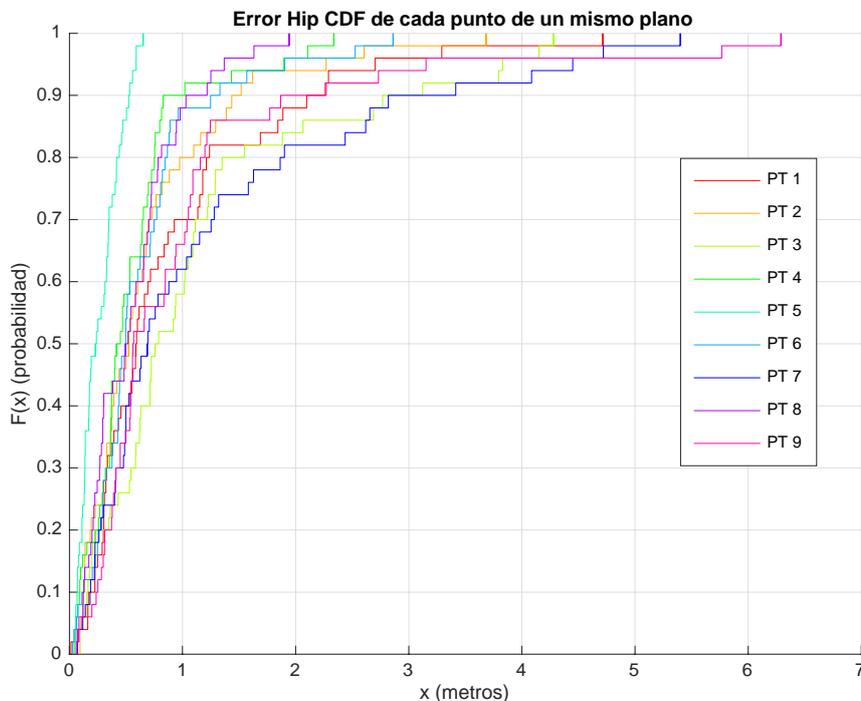


Ilustración 28. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton hiperbólico).

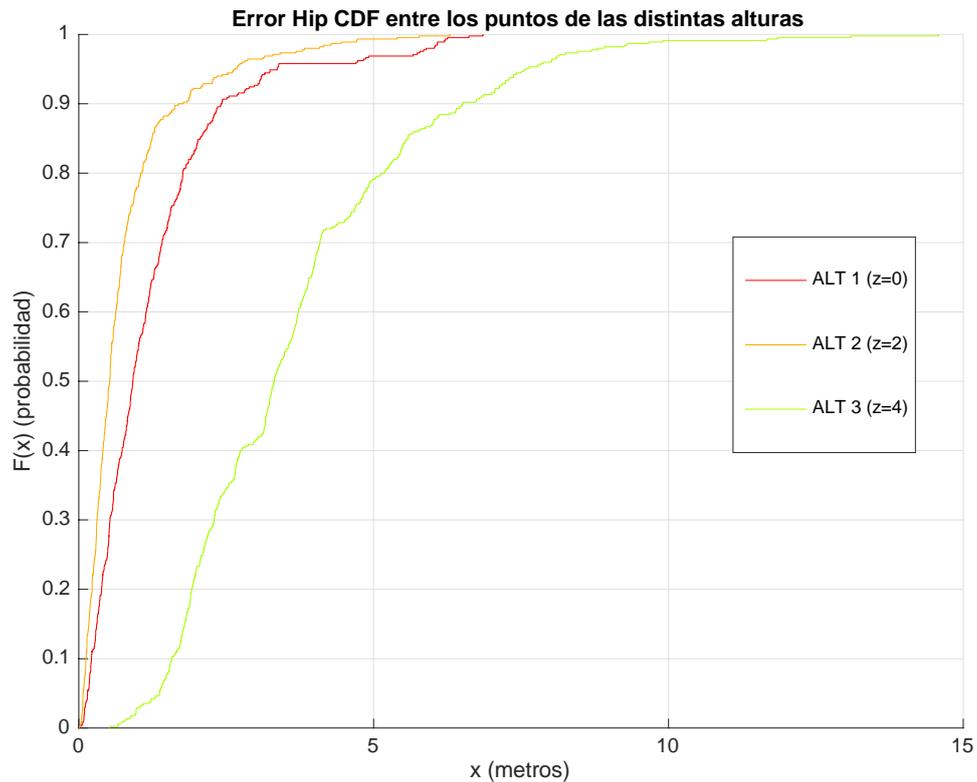


Ilustración 29. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton hiperbólico).

En estos casos anteriores de error, se puede volver a comentar que el error cometido de posicionamiento sigue una misma función de variación tanto si se compara el error de los puntos de un mismo plano, como si se compara el error en los 3 planos diferentes; por lo que se vuelve a concluir que la posición de las balizas no es adecuada para posicionamiento hiperbólico. Aumentando las ilustraciones anteriores, se podrá obtener más información al respecto.

Esta primera ilustración, resultado de la ampliación del error de los puntos calculados en un mismo plano ($z=2$), Ilustración 28, se puede observar que el 90% de todos los puntos de ese plano se posicionan con un error máximo de 3'5 m; mientras que el 100% de los puntos se logra posicionar por debajo de los 6,3 m. Como puede verse comparando órdenes de magnitud, en este caso, se habla de un tamaño de error a tener en cuenta a la hora de posicionar.

Se puede volver a comentar que todas las funciones de cada punto individual siguen una misma distribución de error; pero, los puntos 2, 4, 5, 6 y 8 son los que se posicionan en cabeza con mínimo de error, sobre todo el 5, debido a que se sitúan en la proyección del sistema de balizas.

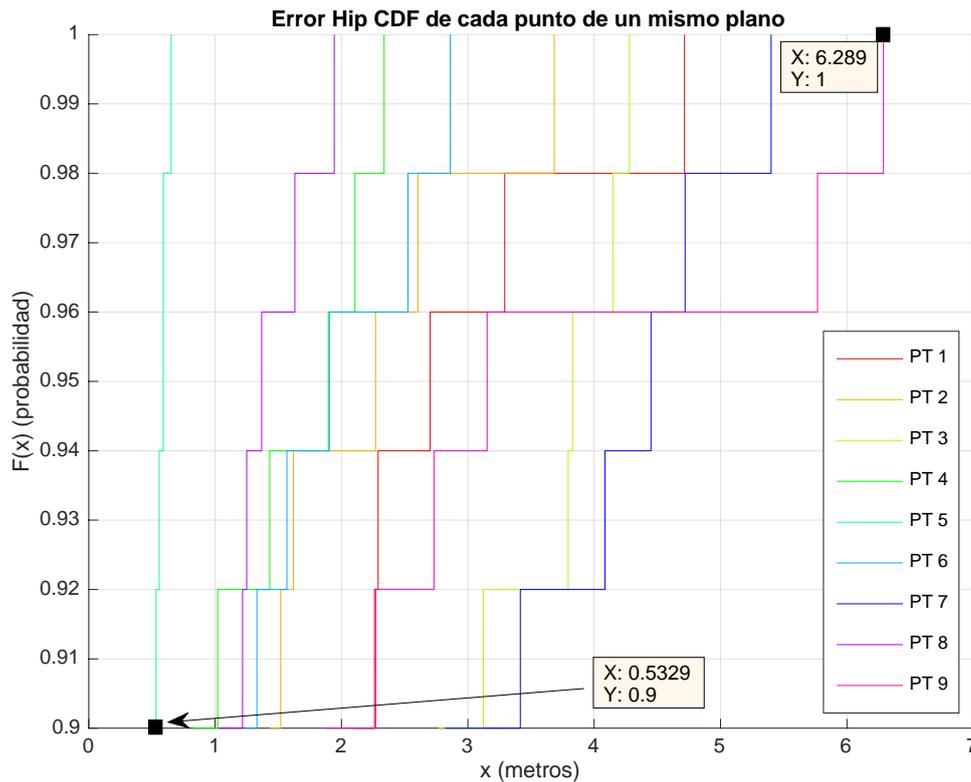


Ilustración 30. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).

En esta segunda ilustración, resultado de la ampliación del error de todos los puntos calculados en los distintos planos, Ilustración 29, se observa que en los dos primeros planos donde se logra calcular mejor la posición, se consigue que el 90% de los puntos se posicionen con un error máximo de 2,5 m. El 100% de los puntos se logra posicionar con un error máximo de 7 m. Esto sigue teniendo unas magnitudes muy grandes para el entorno que se maneja.

En cuanto al tercer plano, es visible en la ilustración que las magnitudes que se manejan son desproporcionadas, al igual que se veía en la ilustración de posicionamiento 3D. Se logra posicionar el 100% de los puntos con un error máximo de 15 m. Esto ya queda fuera del entorno de simulación.

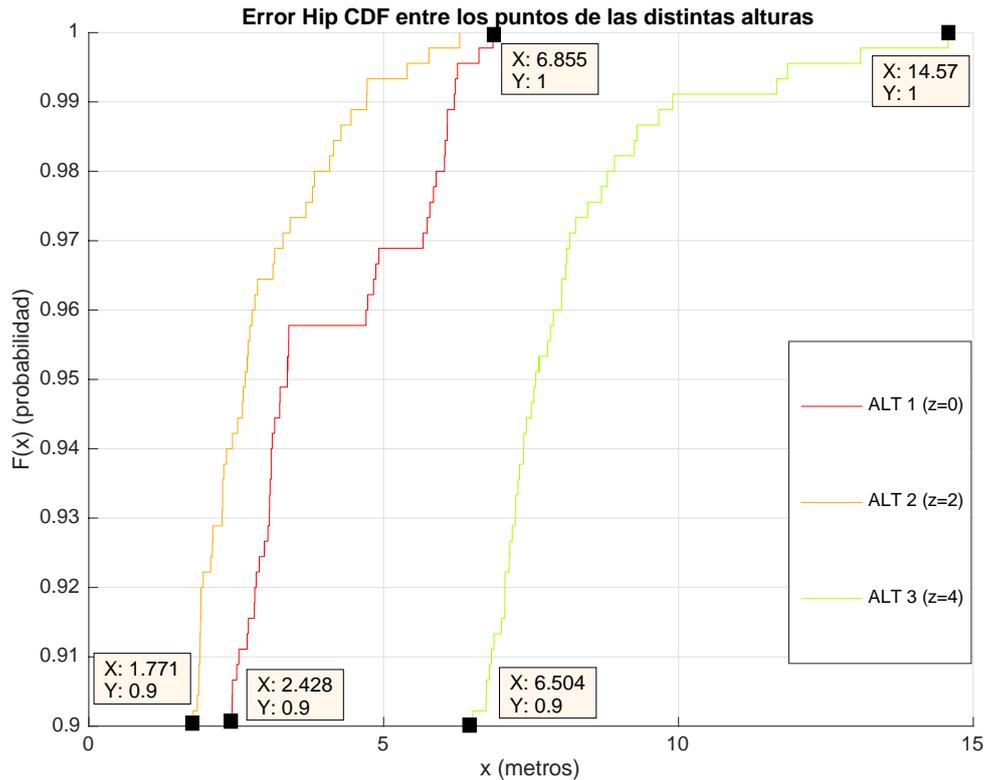


Ilustración 31. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).

6.1.3 PDOP de modo independiente

Con lo comentado anteriormente, se van a calcular los resultados de PDOP para observar la calidad y la precisión de las medidas ya hechas anteriormente.

6.1.3.1 PDOP mediante métodos heurísticos

En esta primera ilustración, se obtienen las medidas de PDOP heurístico de los puntos situados en el plano 'z=2' aplicando el algoritmo de Gauss-Newton esférico. Como se puede observar, los resultados se distribuyen de manera uniforme en círculos con centro la baliza central del sistema de referencia, siendo estos dependientes de la distancia a esa baliza. Estos resultados corroboran lo que ya se había apuntado en el punto anterior. Los puntos que por distancia estén más cerca de las balizas del sistema, se calculan mejor su posición que en el resto de los puntos. Por eso en las esquinas es donde se pierde más la precisión y se obtiene más error. También hay que comentar que el error cometido en toda la ilustración es muy pequeño comparado con el que se verá en otras ilustraciones posteriores.

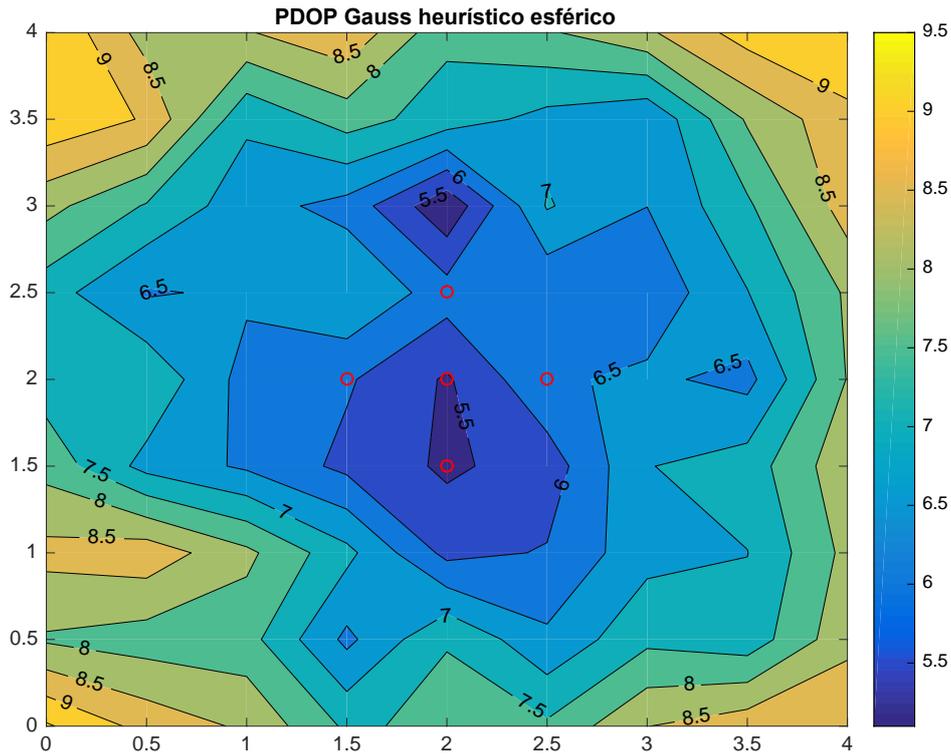


Ilustración 32. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).

En la siguiente ilustración 33, se obtienen las medidas de PDOP de los puntos situados en el mismo plano ' $z=2$ ' aplicando el algoritmo de Gauss-Newton hiperbólico. Observando la ilustración, se vuelve a corroborar que los resultados se distribuyen de manera uniforme en círculos con centro la baliza central, lo que indica que, en los puntos que estén más cerca de las balizas, se obtendrá mejor posicionamiento. En este caso, debido al posicionamiento hiperbólico, se trata con unos errores muy grandes a la hora de posicionar, aunque haya puntos mejores que otros.

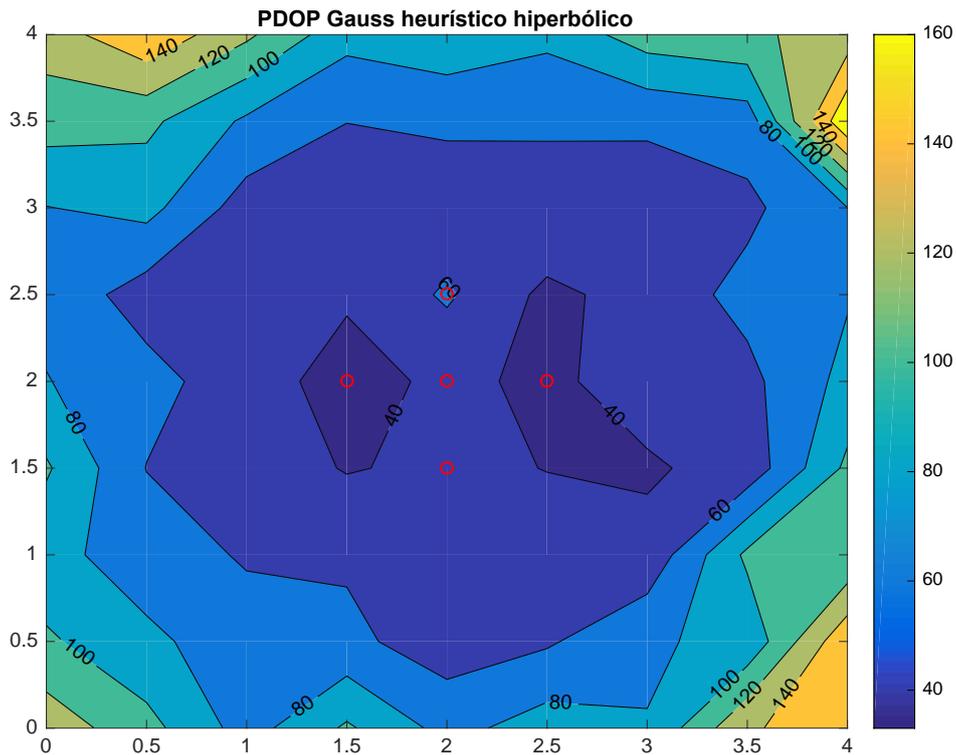


Ilustración 33. PDOP del posicionamiento hiperbólico mediante métodos heurísticos ($z=2$).

Las ilustraciones anteriores, muestran los resultados del PDOP calculados a partir del método heurístico, a través de los cálculos de las varianzas de cada coordenada de cada punto (Ecuación 17). A continuación, se verán los resultados del PDOP a través de otros métodos de cálculo ya explicados en el punto de Análisis de PDOP.

6.1.3.2 PDOP mediante la inversa de la matriz Jacobiana

En las siguientes dos ilustraciones, se representan los resultados de PDOP mediante la inversa de la matriz Jacobiana. De ambas ilustraciones se puede destacar la precisión de los datos en cuanto a uniformidad se refiere, debido a que se clasifica muy bien por circunferencias la precisión que se obtiene en todo en plano. También queda claramente remarcado el hecho de que con el método esférico se obtienen mejores resultados en cuanto a error se refiere que con el método hiperbólico. En ambas ilustraciones vuelve a observarse que el punto colocado debajo de las balizas es el punto en el que se obtiene mejor precisión de posición, mientras que, en las esquinas, es donde se comenten más errores.

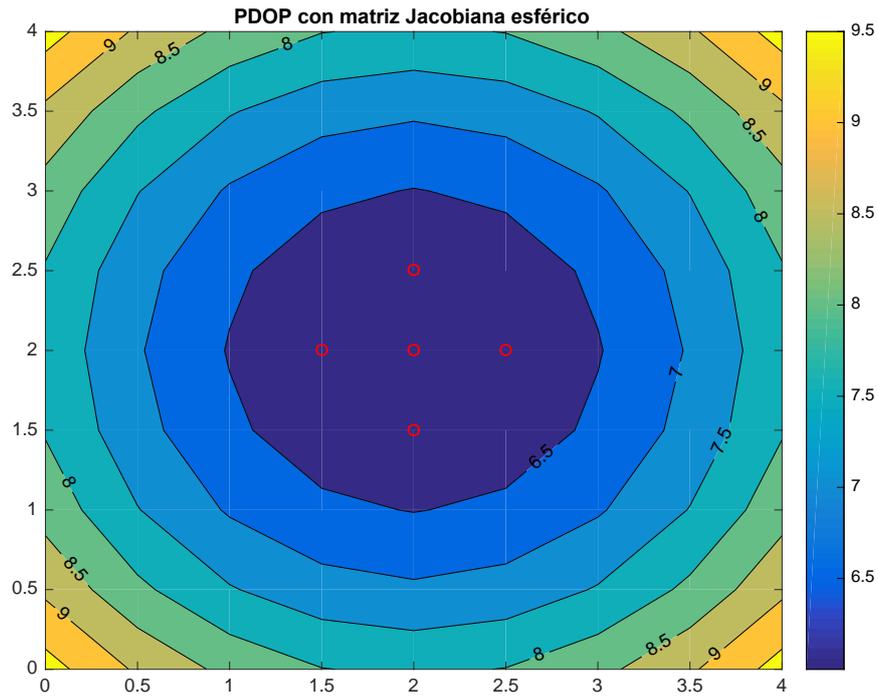


Ilustración 34. PDOP del posicionamiento esférico mediante la inversa de la matriz Jacobiana ($z=2$).

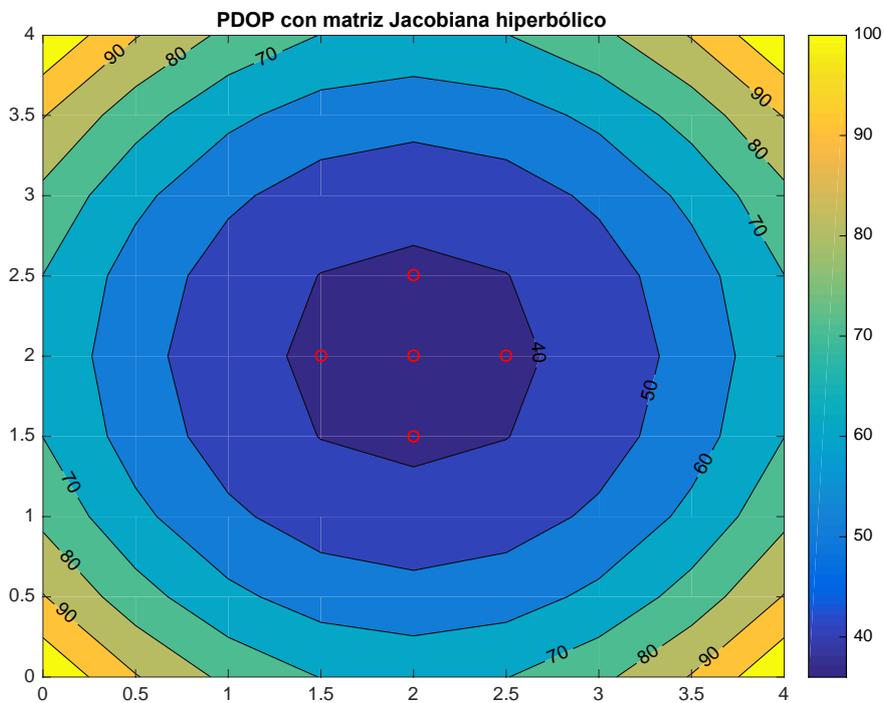


Ilustración 35. PDOP del posicionamiento hiperbólico mediante la inversa de la matriz Jacobiana ($z=2$).

6.1.3.3 PDOP mediante el cálculo de la altura de tetraedros

En la próxima ilustración 36 de cálculo del PDOP, se representa la misma medida con el método de cálculo por alturas a través de distintos tetraedros que se definen mediante las combinaciones de cuatro balizas. A recordar, Ecuación 23. En este caso, el PDOP se obtiene sólo por métodos geométricos, con independencia del algoritmo que posteriormente se implemente.

Se puede observar de la ilustración cómo el PDOP que se obtiene, también sigue una representación a través de círculos concéntricos con punto central el sistema de referencia, lo que indica que los puntos más cercanos a las balizas son los que mejor se pueden posicionar.

Por otro lado, el PDOP tan elevado se debe a que las balizas con las que se trabaja están demasiado cercas las unas de las otras al estar trabajando en la esfera de radio unidad formada a partir del punto de posicionamiento. Si en el sistema de referencia (Ilustración9) las balizas estuvieran más alejadas, al formar la esfera de radio unidad también estarían más separadas las unas de las otras, lo que permitiría que la precisión fuera mejor que la mostrada.

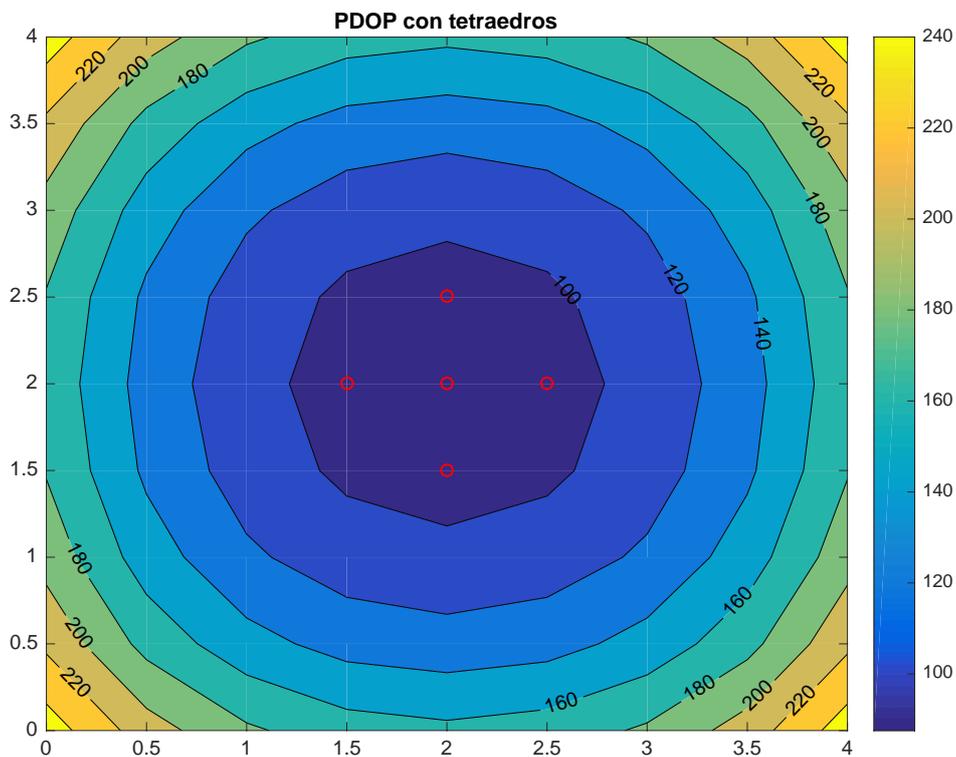


Ilustración 36. PDOP de posicionamiento mediante el cálculo de la altura de tetraedros ($z=2$).

6.1.3.4 PDOP mediante el Bideterminante de Cayley-Menger

Con esta técnica, se vuelven a utilizar las alturas de los tetraedros para la obtención de las medidas de PDOP en el plano ' $z=2$ '. En este caso, se utilizan las distancias entre las balizas utilizadas para la formación de los tetraedros, para formar los determinantes que nos llevan al cálculo de la altura; así con este método se evita el uso de funciones trigonométricas.

Como se puede observar de la siguiente ilustración 37, lógicamente se llega a los mismos resultados debido al uso del mismo método de la altura (aunque matemáticamente se haya llegado a ello por dos vías distintas).

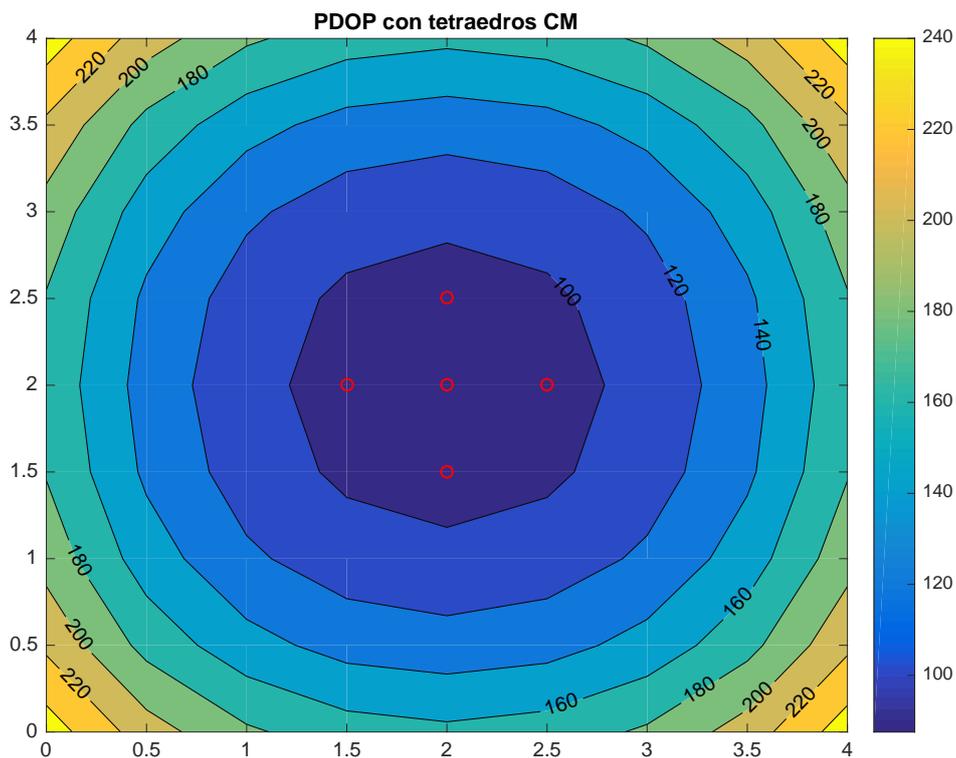


Ilustración 37. PDOP de posicionamiento mediante el Bideterminante de Cayley-Menger ($z=2$).

6.2 Algoritmo de fusión

Una vez que se han visto los resultados que se obtienen de aplicar el modelo independiente al método de Gauss-Newton, se puede aplicar al mismo modelo una ampliación de este teniendo en cuenta 3 sistemas de balizamiento (3 LPSs diferentes en planos perpendiculares) en vez de uno. Con esto se puede mejorar esa distorsión que se tenía a la hora de posicionar debido a la situación del sistema de referencia empleado. Una vez se tengan todos los datos obtenidos, se procederá a aplicar el algoritmo de fusión de datos como se ha ido comentando paso a paso anteriormente.

En la siguiente ilustración 38 se puede visualizar cómo se tienen los 3 sistemas de balizas posicionados en los distintos planos, y los resultados de posición que se obtienen de cada uno de ellos marcados en el mismo color que el sistema de balizas al que pertenecen.

Los resultados provenientes de cada sistema de balizas son los obtenidos de aplicar el método esférico correspondiente al algoritmo de Gauss-Newton, que son los expuestos en los resultados anteriores comentados en el apartado de modo independiente.

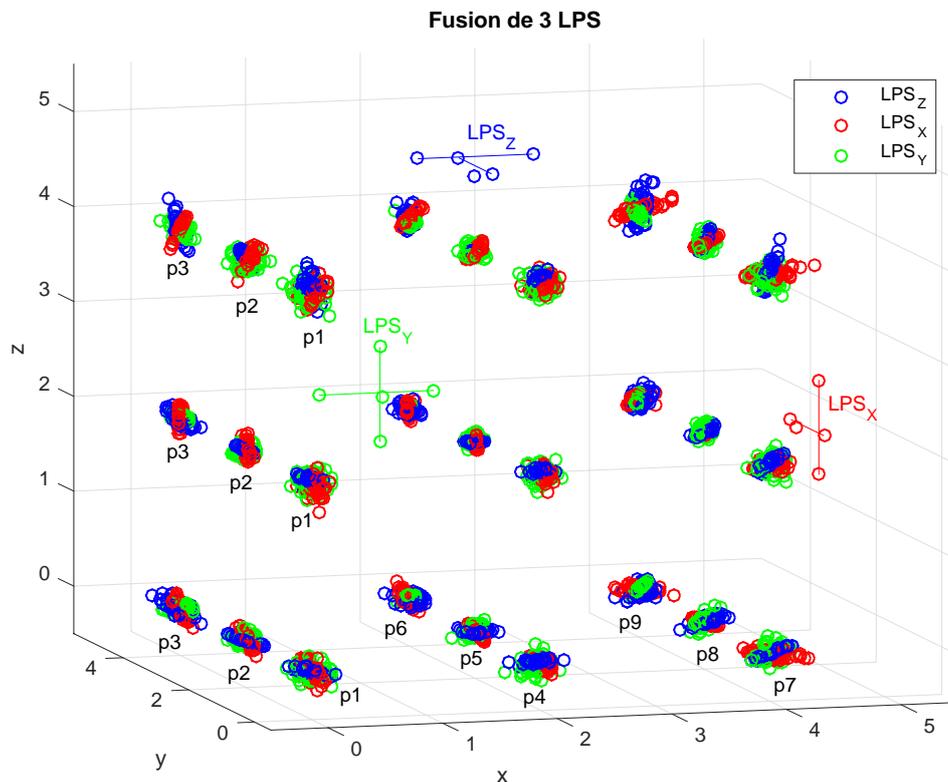


Ilustración 38. Posicionamiento en 3D con el método Gauss-Newton esférico a través del uso de 3 sistemas de referencia distintos.

En la siguiente ilustración, se muestra únicamente el resultado de la fusión completa de todos los datos. A simple vista se puede percibir que las nubes de puntos son de un tamaño similar, se verá con más detalle posteriormente, pero, se puede decir que el método de fusión corrige más

esa dispersión de posición que se tenía; por lo que permite acotar más los límites de error que se tenían a la hora de posicionar cierto objeto.

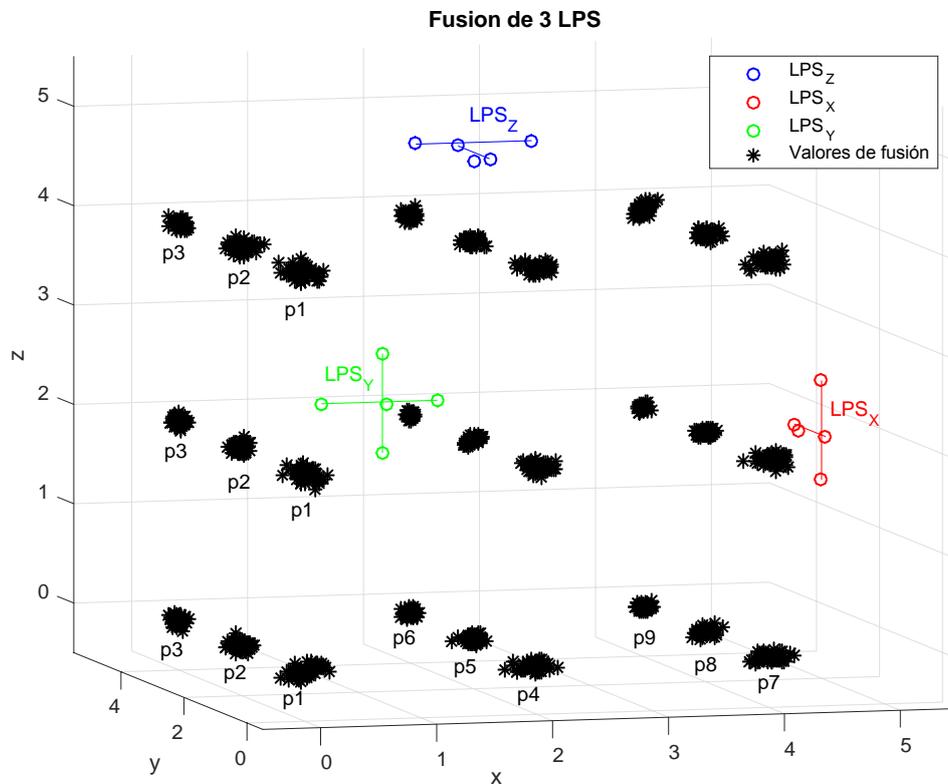


Ilustración 39. Posicionamiento 3D a través del método de fusión MLE.

En la siguiente ilustración 40 se observan tanto los elipsoides de error como sus proyecciones. Primero se observa la proyección en las paredes del cubo, que es lo que se considera el entorno de simulación, donde se aprecia con más detalle que las elipses son de tamaño similar. En la siguiente ilustración también se aprecia mejor que en el plano 'z=2' es donde se posicionan los puntos con menos error puesto que las proyecciones son algo más pequeñas.

En segundo lugar, se observará con más detalle la proyección del eje 'z' que se tiene de los elipsoides calculados en las nubes de puntos centrales. En esa ilustración también se puede apreciar cómo la proyección en el eje 'z' mejora con respecto al método anterior puesto que la fusión de los datos reduce más error en el posicionamiento.

La circunferencia de color azul corresponde al punto 'p5' del plano 'z=0', la circunferencia de color magenta corresponde al punto 'p5' del plano 'z=2', y la circunferencia de color rojo corresponde al punto 'p5' del plano 'z=4'.

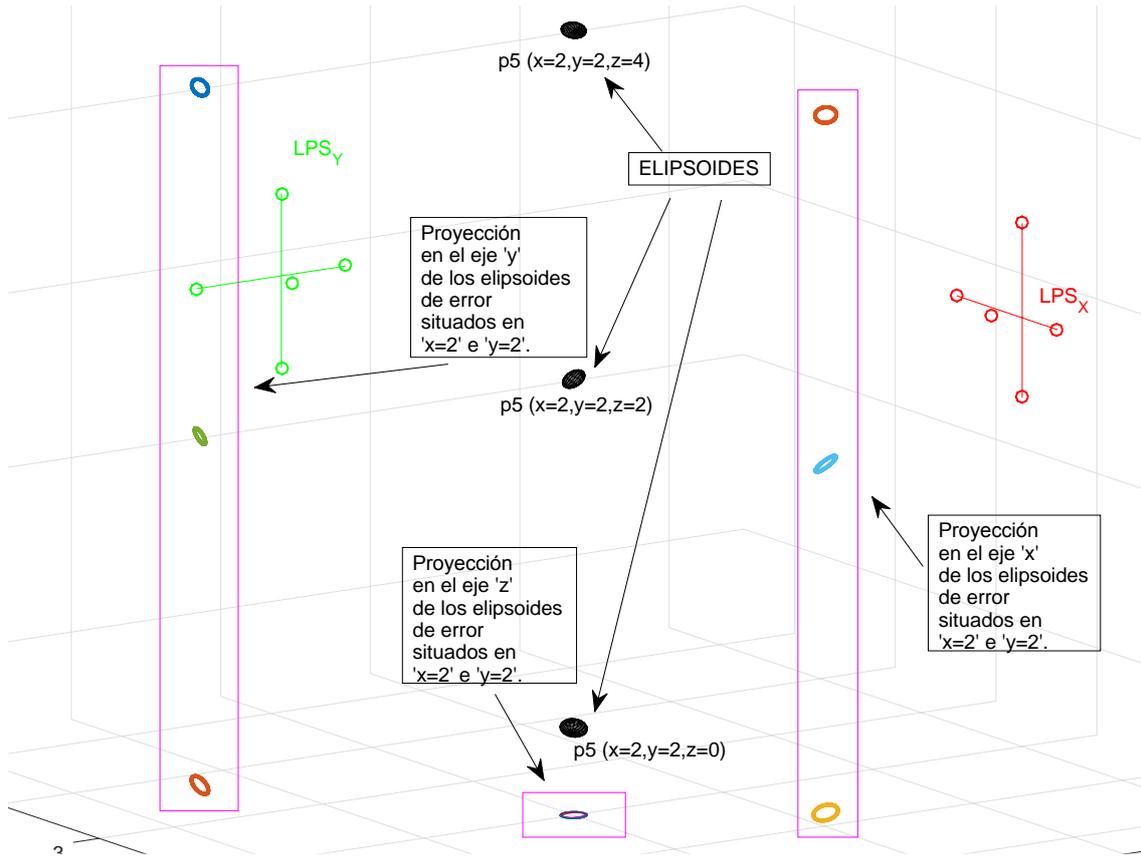


Ilustración 40. Proyecciones en todos los planos de la Ilustración 39.

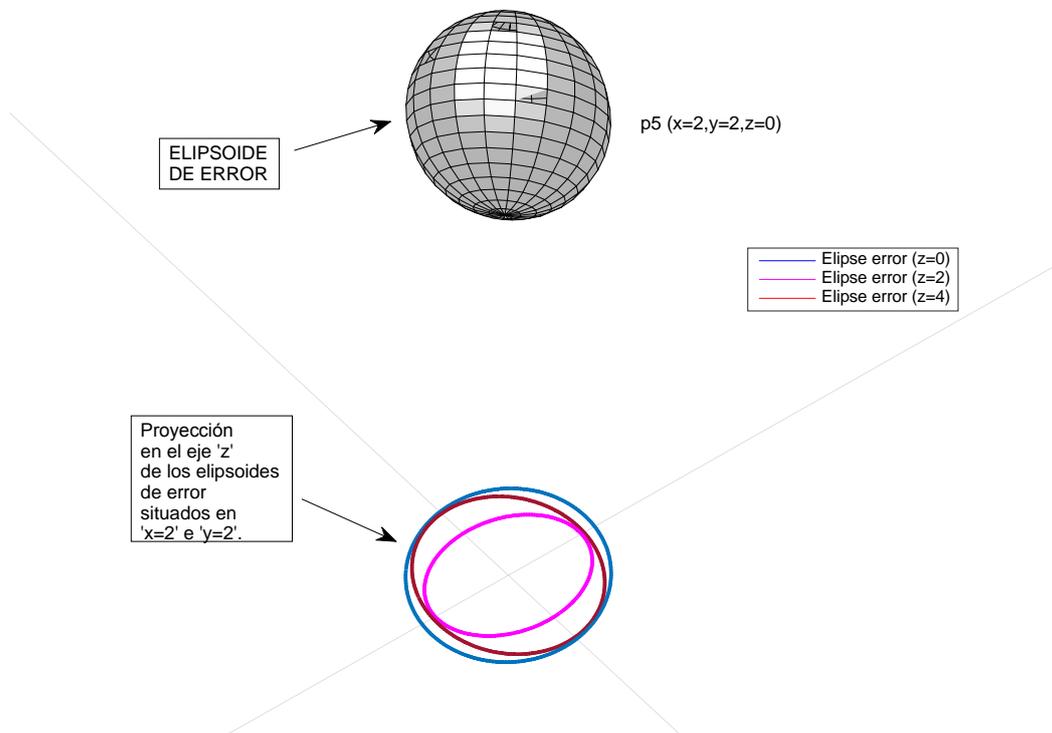


Ilustración 41. Elipsoide de error y proyecciones en el plano 'z' de la Ilustración 40.

En cuanto al error cometido en ese proceso de fusión, cabe destacar que, al igual que en casos anteriores, las funciones de error siguen una distribución muy similar donde el error cometido vuelve a ser muy pequeño (centimétrico) comparado con otros vistos.

Se puede observar también de las siguientes ilustraciones que, debido al proceso de fusión, la situación de los puntos en la gráfica, en cuanto a error se refiere, ya es indiferente de donde se sitúen las balizas debido a que se han mezclado y rotado las matrices de datos relativas a un único sistema de posición. Por lo que se puede apreciar, ahora los mejores puntos con mínimo error son los que se sitúan más cerca de los tres sistemas de balizas, por eso, los puntos que encabezan la gráfica son los puntos 'p9', 'p8' y 'p6'.

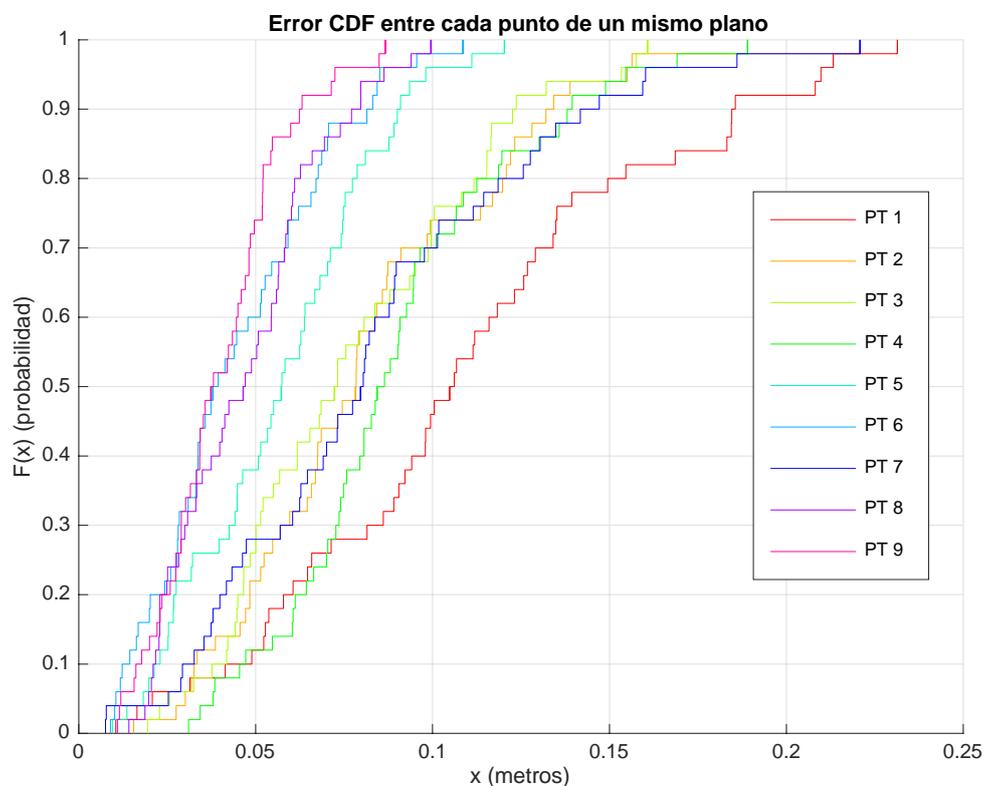


Ilustración 42. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Fusión).

La siguiente ilustración, resultado de la ampliación de la anterior, se puede obtener que el 90% de los puntos de ese plano en particular se posicionan con un error inferior a 19 cm, y el 100% de los puntos se posicionan con un error inferior a 24 cm. El mejor punto de ese plano logra posicionarse con un error menor de 10 cm. Otra vez se vuelve a hablar de unos errores con unas magnitudes muy ínfimas en comparación con el tamaño de la habitación de simulación.

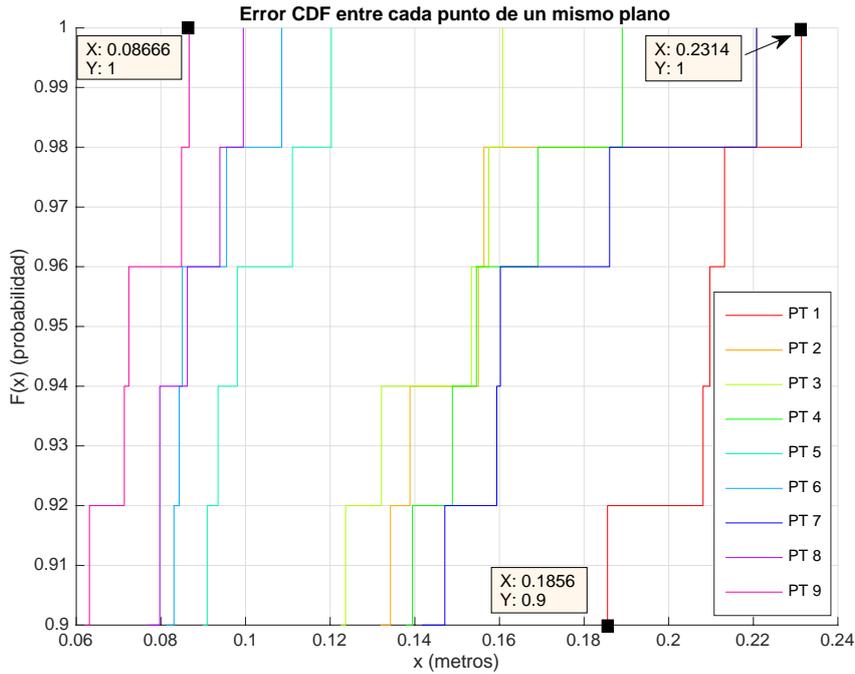


Ilustración 43. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Fusión).

En la siguiente ilustración, se puede comentar lo dicho anteriormente. Se visualizan unos órdenes de magnitud de acuerdo con el sistema, y todas las funciones de los distintos planos se distribuyen de manera similar.

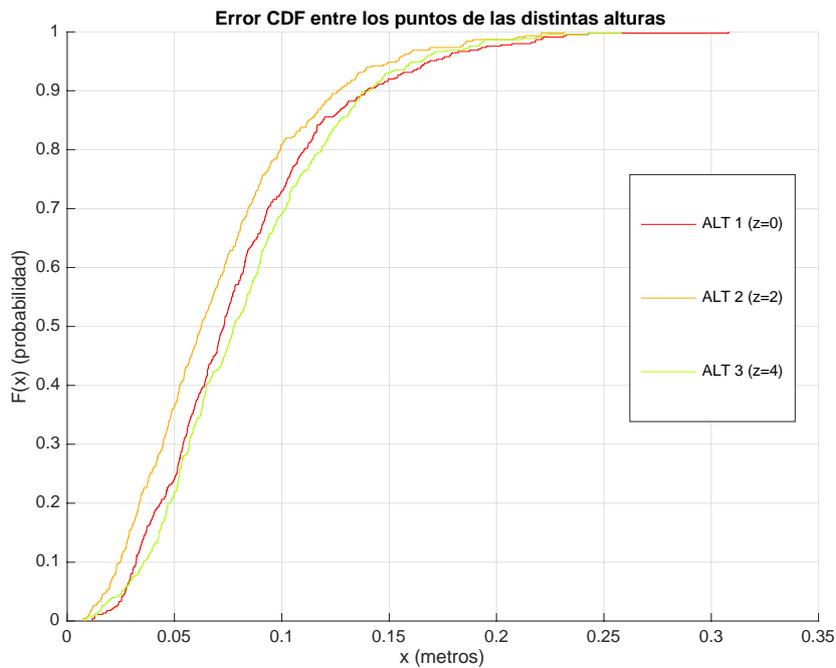


Ilustración 44. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Fusión).

En la ilustración ampliada de a continuación, que puede visualizar que el 90% de los puntos se posicionan con un error menor de 15 cm de distancia con respecto al punto real; mientras que el 100% de los puntos se posicionan con un error máximo de 30 cm. Cabe destacar que los 3 planos van prácticamente a la par en cuanto a error cometido; aunque aquí se puede volver a visualizar que el plano 'z=2' lleva la delantera del mínimo error.

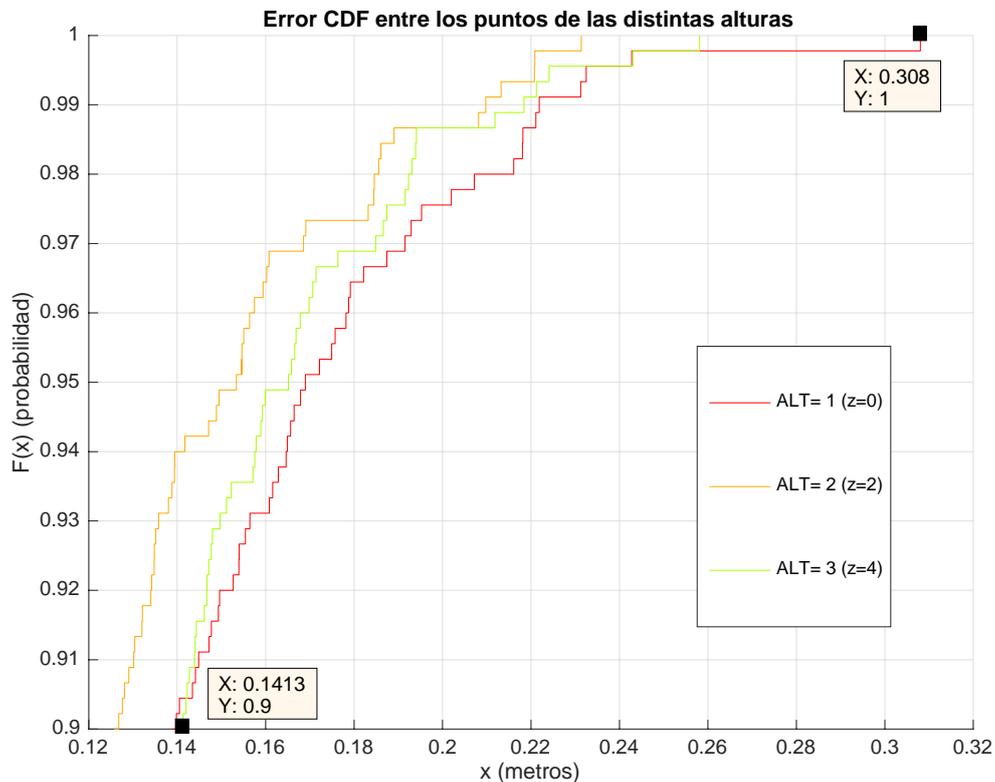


Ilustración 45. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Fusión).

No se ha probado el método de fusión con los datos hiperbólicos calculados del modo independiente debido a la dispersión que tienen algunos datos. Es más relevante y visible los resultados con los datos esféricos.

6.2.1 PDOP de fusión

Como complemento a los resultados anteriormente comentados a través de la fusión de los datos obtenidos del modo independiente, se van a mostrar los resultados del cálculo del PDOP obtenido para el algoritmo de fusión.

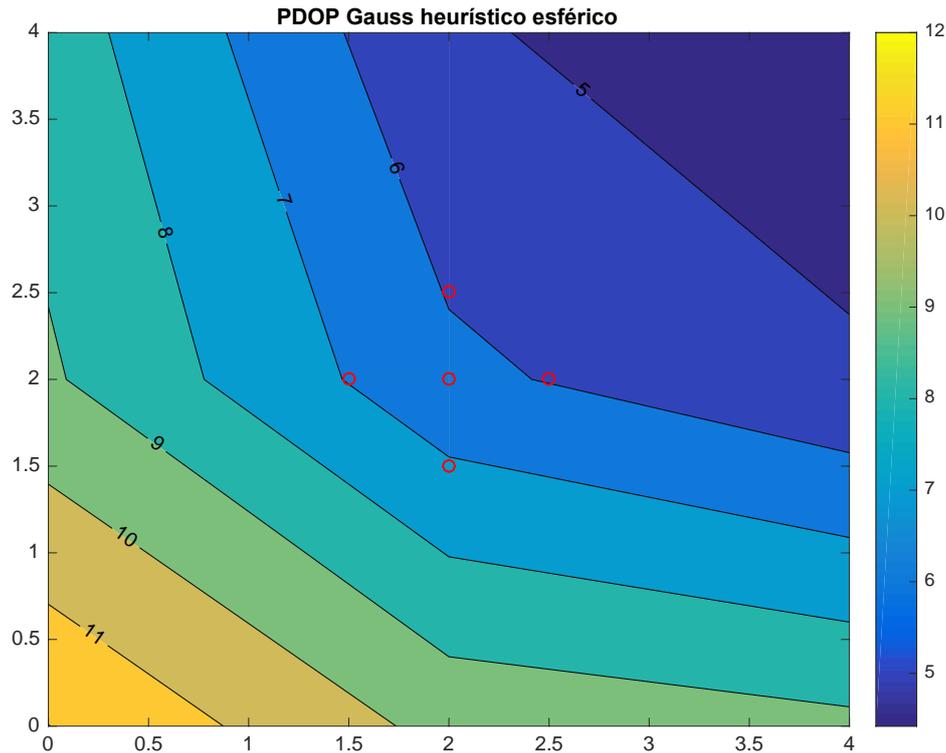


Ilustración 46. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).

Comparando los resultados obtenidos en la Ilustración 46 con los resultados de la Ilustración 32, que son los resultados correspondientes al mismo método de cálculo; se puede observar que en ambos casos la precisión de los datos es la misma, por lo que se estima un buen posicionamiento y fusión de todos los datos. Cabe destacar que, aunque el rango de la precisión sea la misma, el mapa de representación de los datos es diferente, y esto es algo muy indicativo. Al tener los datos obtenidos del modo independiente, la representación del mapa del PDOP gira en torno a las balizas puesto que en los puntos más cercanos a ellas se estima una mejor precisión de aquellos datos. Esta misma conclusión se puede aplicar al caso del PDOP de fusión aquí representado. En este caso se parte de 3 sistemas de balizas colocados en los planos ' $x=5$ ', ' $y=5$ ' y ' $z=5$ '; por lo que, siguiendo la misma teoría anterior, la mejor precisión de posicionamiento se obtiene en aquellos puntos que se sitúen lo más cerca posible de las balizas, y eso es justo lo que se representa en la Ilustración 46.

El resto de formas para el cálculo o estimación del PDOP no se van a tener en consideración en este punto debido a que dependen del propio punto donde se calcula el posicionamiento y no de la combinación de los datos obtenidos del modo independiente.

6.3 Modo simultáneo

En este modo, se calcula directamente una posición estimada a través de los 3 sistemas de balizamiento (LPSs) colocados en los distintos planos ya comentados, como si fuese un solo LPS de más balizas, esto es, suponiendo que los sistemas de balizas se manejan simultáneamente para obtener hasta 15 distancias y a partir de ellas aplicar los algoritmos de posicionamiento. El

cálculo se recuerda, en este caso, que se hace de manera simultánea con las 15 balizas al mismo tiempo. Eso provoca que se hallen mejores precisiones en los resultados.

6.3.1 Trilateración esférica

Las siguientes ilustraciones muestran directamente el posicionamiento a través del algoritmo en el entorno ya conocido. Se puede observar que el posicionamiento es muy bueno debido a que se puede diferenciar cada cálculo de posición en cada punto con una dispersión mínima por lo que se puede apreciar en las ilustraciones.

Se va a mostrar cada plano de manera independiente para que se aprecie mejor lo comentado.

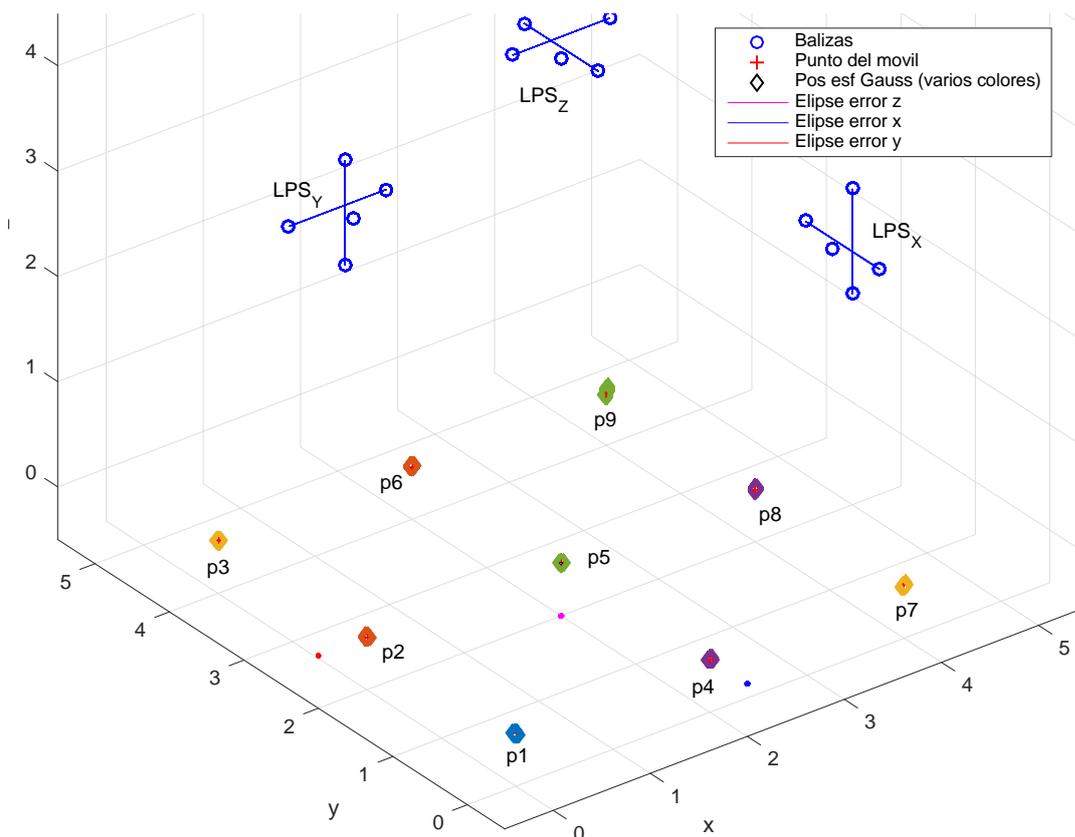


Ilustración 47. Posicionamiento en 3D con el método Gauss-Newton esférico ($z=0$).

Como se puede apreciar en la ilustración, no parece que haya dispersión en ningún sentido en particular, pero puesto que el error de las elipses es mínimo, se comprobará en posteriores ilustraciones para corroborar lo comentado.

En la siguiente ilustración se muestra el siguiente plano de posicionamiento donde se vuelve a corroborar a simple vista lo ya expuesto anteriormente. Como se puede ver, no se aprecia si el error mejora algo con respecto al anterior plano puesto que no se aprecia debidamente el tamaño de las elipses y el posicionamiento de los puntos es muy semejante al anterior.

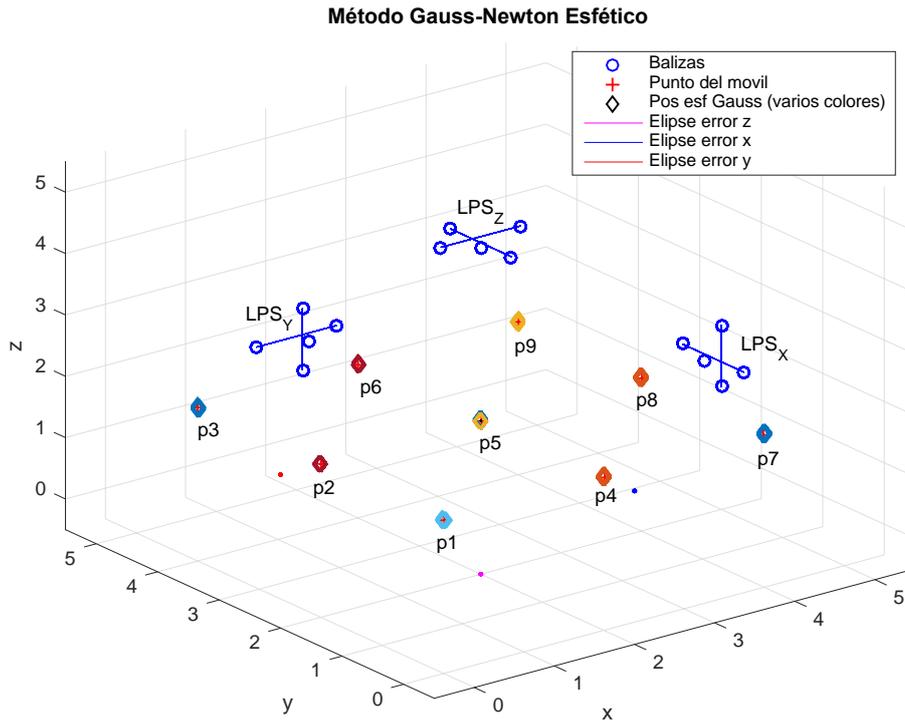


Ilustración 48. Posicionamiento en 3D con el método Gauss-Newton esférico (z=2).

A continuación, se muestra el último plano de posicionamiento donde cabe comentar lo ya dicho anteriormente.

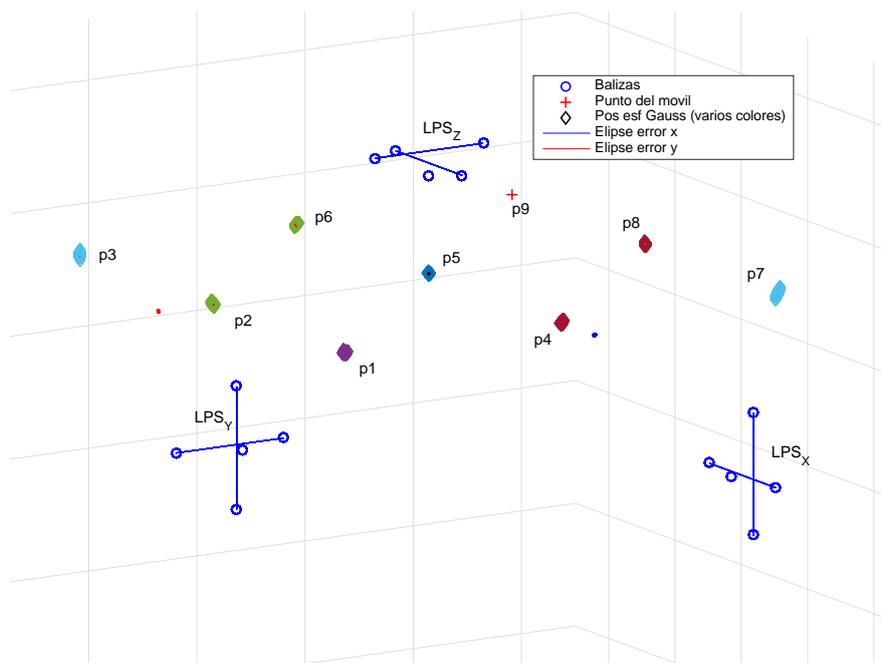


Ilustración 49. Posicionamiento en 3D con el método Gauss-Newton esférico (z=4).

Se puede concluir que, a priori parece que el posicionamiento es mejor que el que se presentó para el modo independiente, y que, parece que se puede hablar de unos errores mucho menores a la hora obtener posiciones. Esto se corroborará mejor en próximas ilustraciones donde se podrán observar mejor los resultados.

En esta ilustración 50, se observa que los elipsoides de error quedan cubiertos por las nubes de puntos, lo que significa que todos los puntos correspondientes al cálculo de la posición estimada se corresponden con mínimo de error al punto real asignado al que se le calcula la posición. De las proyecciones de esos elipsoides, sólo mencionar que con respecto al resto de escenarios generados son más pequeñas, lo que implica una reducción del error; y que son similares en todos los puntos calculados. Esto implica que, con este método, no hay tanta diferencia entre el cálculo de posiciones en distintos planos como podría haberla en los métodos anteriores.

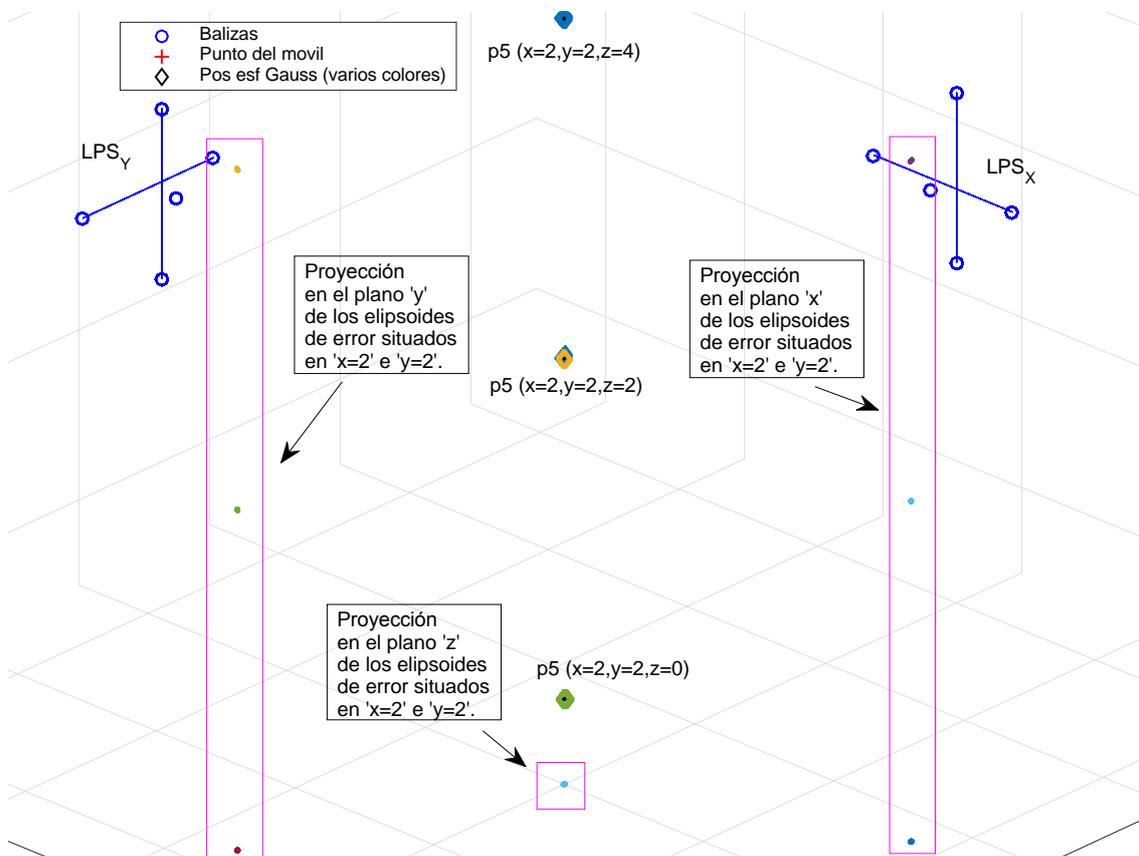


Ilustración 50. Proyecciones en todos los planos de la Ilustración 47, 48 y 49.

En esta ilustración posterior, se corrobora mejor cómo las proyecciones en el eje 'z' de los distintos planos donde se sitúan los puntos a calcular son muy semejantes.

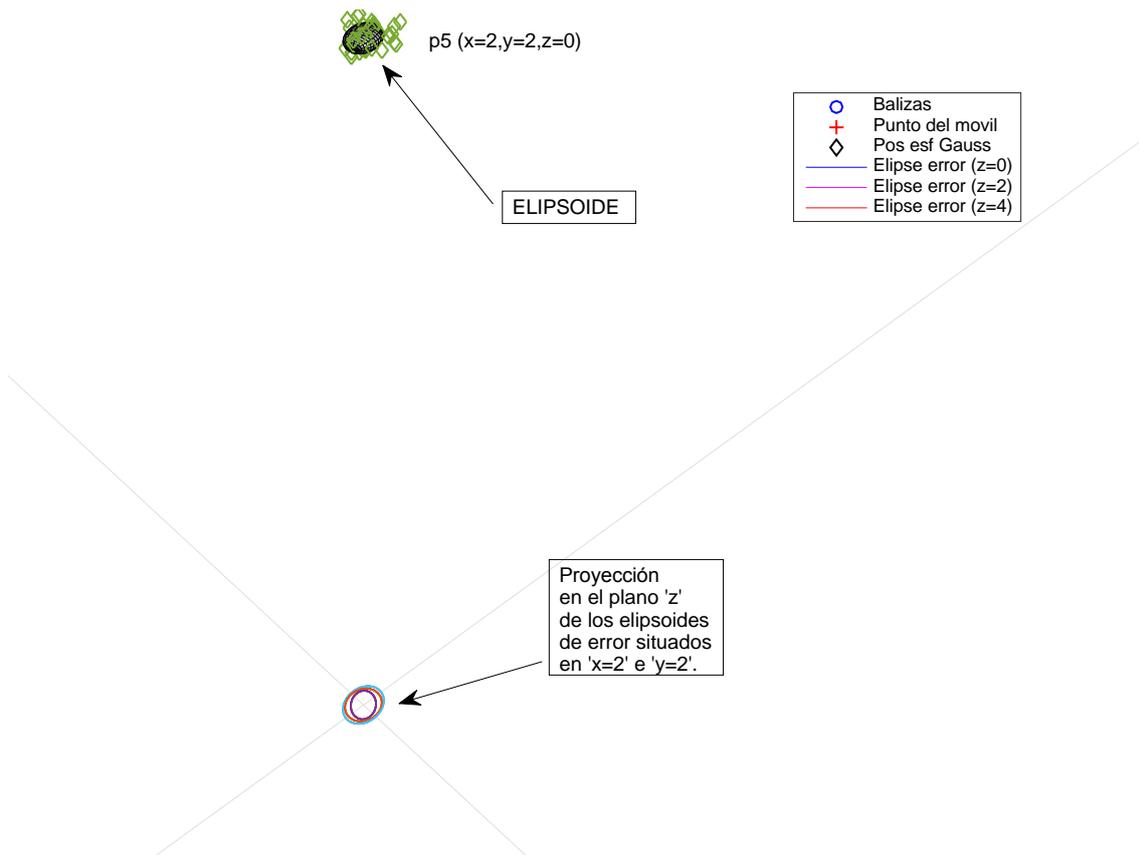


Ilustración 51. Ejemplo de elipsoide de error y proyecciones en el plano 'z=0' de la Ilustración 47, 48 y 49.

En las siguientes ilustraciones se muestra el error cometido en el cálculo de cada punto estimado en el plano 'z=2', y el error cometido como la suma de todos los puntos en cada plano de manera independiente. De estas gráficas cabe comentar que varían de la misma manera, lo que supone que el sistema de referencia permite calcular todos los puntos, con más o menos error, pero los permite calcular; con la peculiaridad, como ya se ha mencionado, que hay un poco menos de error en los puntos que se sitúan bajo la posición de las balizas. Esta situación es visible en la primera ilustración donde, la función con menos error corresponde al punto 5, que es el punto que se sitúa debajo de la baliza central; y a éste le siguen los puntos 2, 4, 6 y 8 que son los que se sitúan bajo la proyección del resto de balizas. En los puntos correspondientes en las esquinas es donde se encuentra más error.

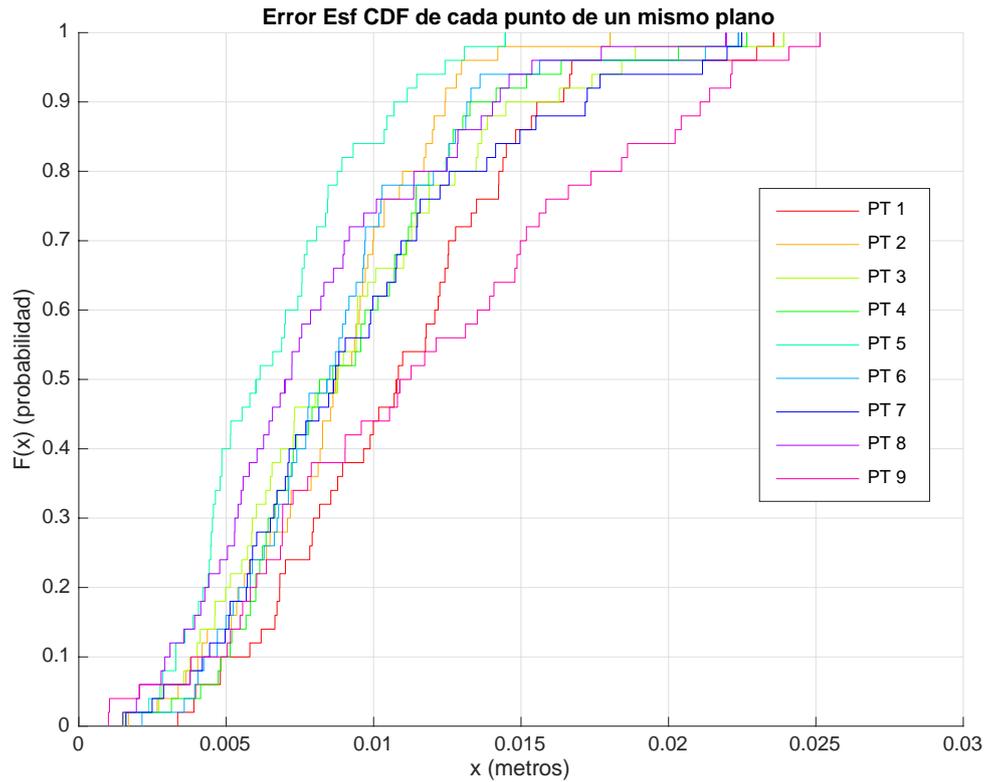


Ilustración 52. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton esférico).

En la siguiente ilustración 53, que es un aumento de la anterior, se puede ver que el 90% de los puntos se posicionan con un error menor de 2'2 cm. El 100% de los puntos se posicionan con un error menor de 2'5 cm. El mejor de los puntos, que vuelve a ser el que se sitúa debajo de la baliza central, punto 5, se logra posicionar con un error máximo de 1'45 cm. Se vuelve a visualizar que todas las funciones de error de cada punto se distribuyen de manera similar; y que en los puntos de las esquinas es donde se acumula más error.

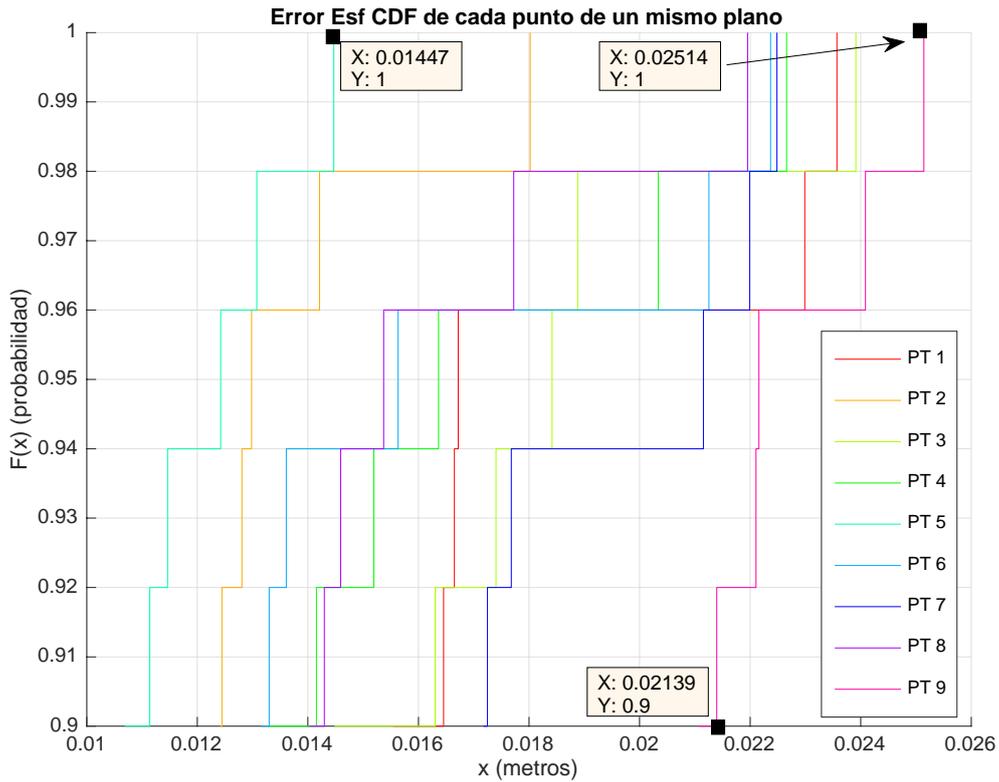


Ilustración 53. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton esférico).

En la siguiente ilustración, se puede apuntar que hay una pérdida muy grande de precisión debido a que hay un punto que el algoritmo no es capaz de posicionar, este es el que corresponde con el último punto, punto 9, del último plano, 'z=4'. De ahí el problema que ocurre con la gráfica que se presenta en la siguiente ilustración con el color verde. Si se obvia este detalle, se puede ver que el error cometido en todos los planos es mínimo y se encuentra uniformemente distribuido.

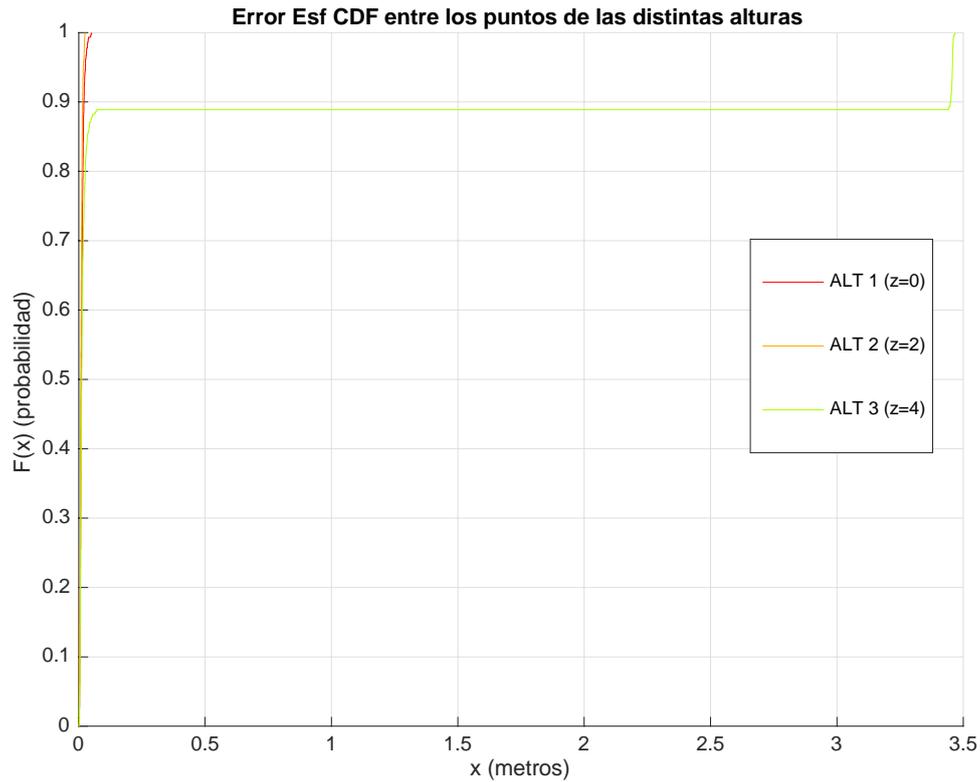


Ilustración 54. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton esférico).

De la siguiente ilustración, que es un aumento de la anterior, se verifica que el 89% de los puntos del tercer plano se calculan con un error máximo de 7,3 cm junto al resto de puntos de los otros dos planos. Debido al pequeño error que sufre el algoritmo, el resto de los puntos se obtienen con un error de 3,5 m. Esto es debido a la falta de ese único punto, que cambia la función del error.

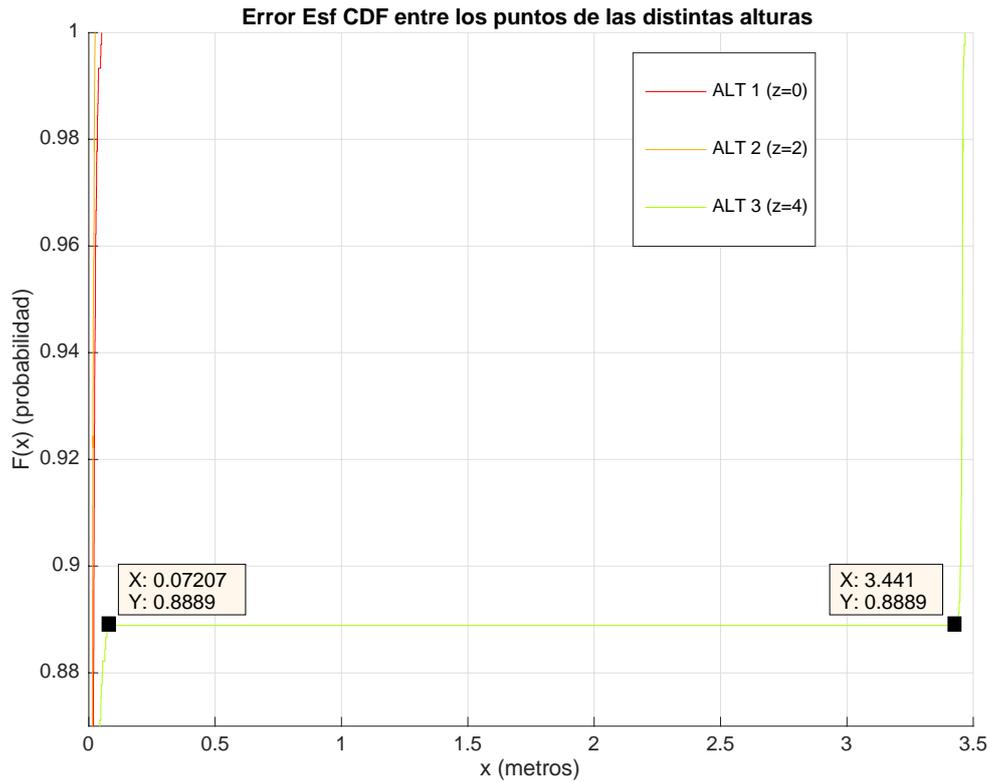


Ilustración 55. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico). Parte I.

En la siguiente ilustración 56, que continúa siendo parte de la misma ilustración, se puede ver que el 100% de los puntos en los dos primeros planos se calcula con un error máximo de 5 cm, y se ve como el segundo plano es ligeramente mejor que el primero, lo que significa que en el cálculo de los puntos del segundo plano se comete un poco menos de error, lo que sugiere que el sistema de balizas le viene mejor.

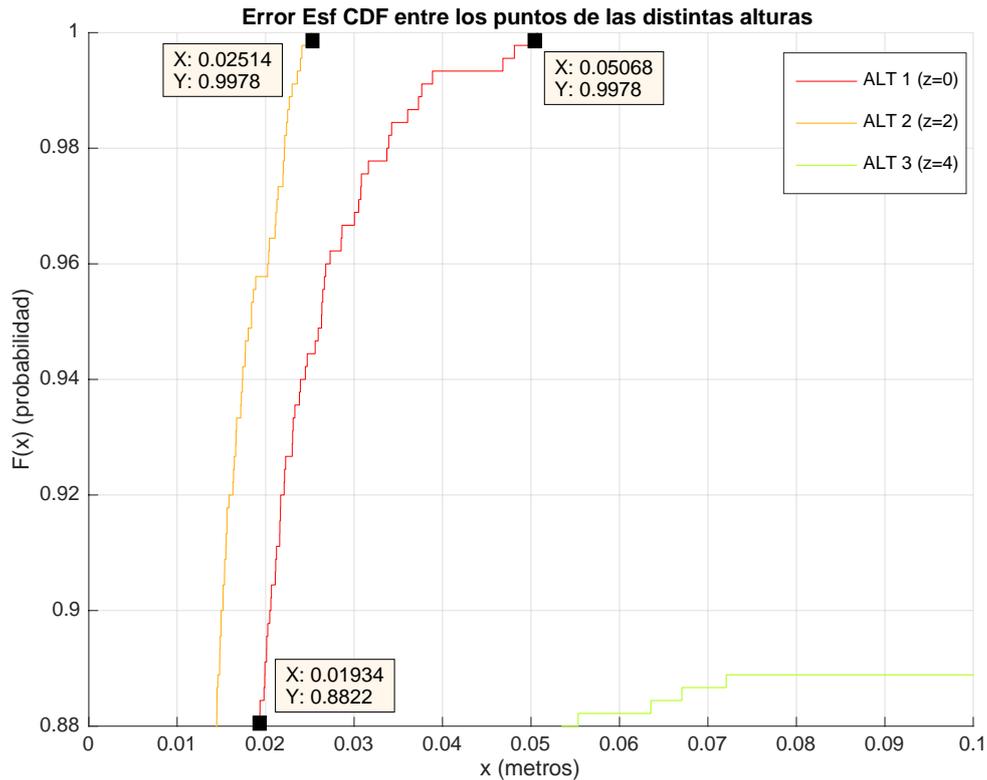


Ilustración 56. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton esférico). Parte II.

6.3.2 Trilateración hiperbólica

Después de analizar el método esférico, se procederá a analizar el método hiperbólico. En las siguientes ilustraciones, se vuelve a encontrar el posicionamiento en 3D de varios puntos en un espacio a través del método hiperbólico. Se puede apreciar que, en los dos primeros planos de cálculos, ' $z=0$ ' y ' $z=2$ ', se logra mejorar el posicionamiento en los puntos indicados por cruces comparado por lo contemplado en los resultados del modo independiente. En este caso, se logra disminuir en gran medida el error cometido al posicionar salvo en el plano ' $z=4$ '. Aquí, vuelve otra vez el problema que se tenía desde un principio. Se debe encontrar alguna indeterminación a la hora de calcular los puntos de ese plano debido quizás a la cercanía de las balizas. Es por eso que los últimos puntos no se pueden posicionar de manera correcta. Todo esto se puede encontrar en la siguiente ilustración.

También se puede observar en las siguientes ilustraciones los elipsoides y sus proyecciones en todos los planos. Al igual que el posicionamiento de los puntos, en los dos primeros planos los elipsoides en los puntos centrales quedan cubiertos por las nubes de puntos correspondientes; por lo que se asume que, en esas posiciones, los puntos estimados han sido bien calculados. Esto lo verifican también las proyecciones de esos elipsoides, donde se aprecia un mínimo de error en esos puntos. Cabe indicar que, en los puntos en los que se calculan los elipsoides, son los puntos que se encuentran debajo de la baliza central; por lo que son los puntos que mejor están posicionados. Independientemente de esa ventaja, los puntos de los dos primeros planos se posicionan con bastante éxito.

En el tercer plano, es donde se verifica el fallo del algoritmo. Debido a la dificultad de posicionar un punto en ese plano, el algoritmo se inicializa en un punto inicial correspondiente a la esquina inferior de la ilustración que es donde se acaban posicionando todos los puntos de ese plano debido, otra vez, a la imposibilidad de encontrar un punto mejor. Por tanto, de las siguientes ilustraciones, es visible también como es el error de posicionamiento del punto central del último plano.

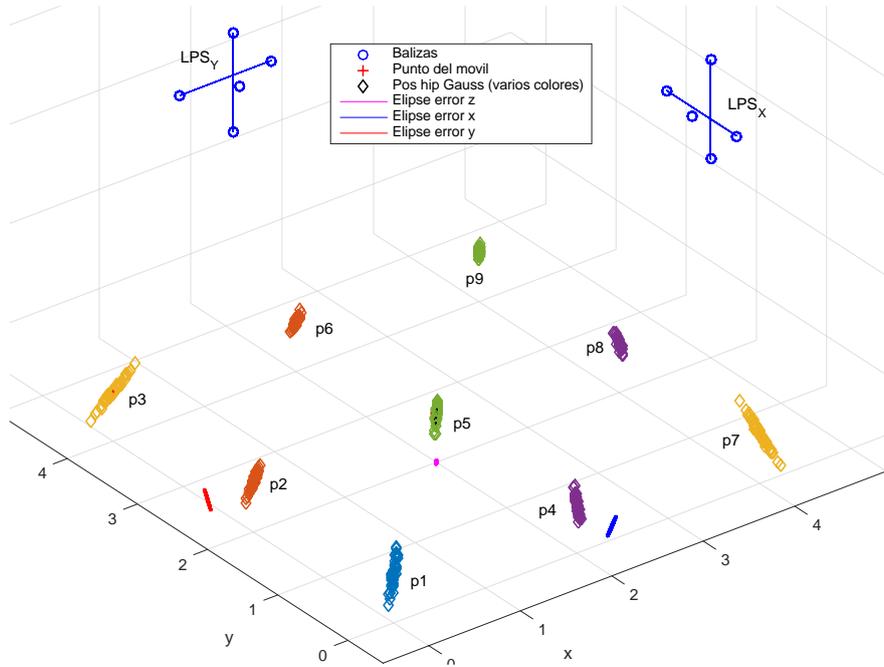


Ilustración 57. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=0$).

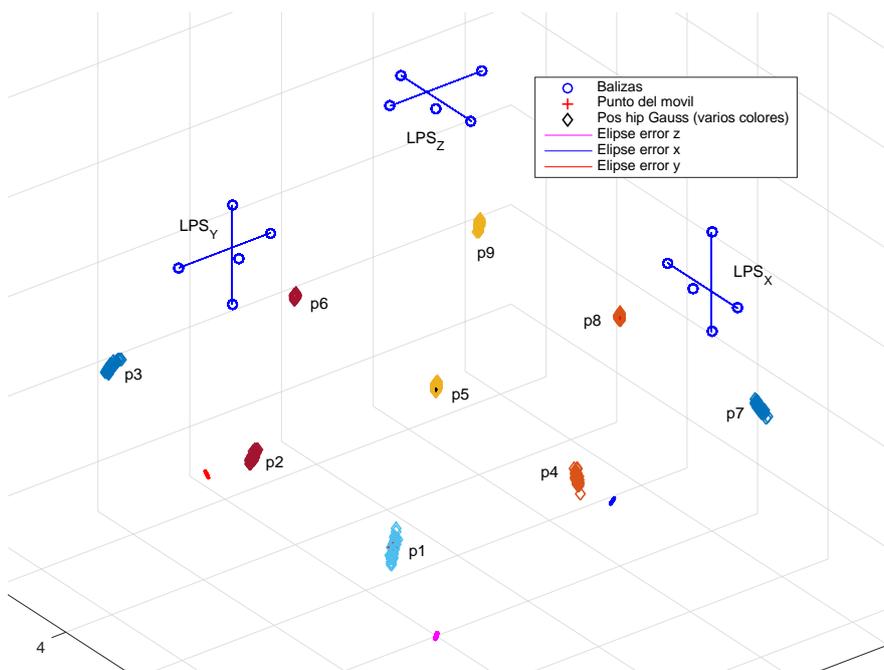


Ilustración 58. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=2$).

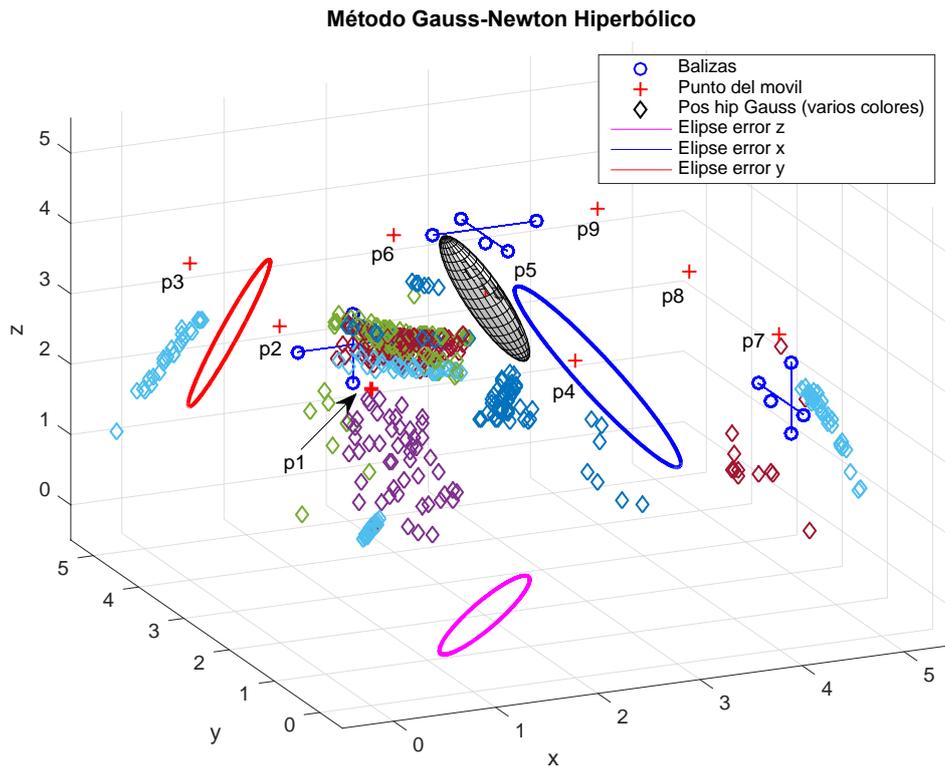


Ilustración 59. Posicionamiento en 3D con el método Gauss-Newton hiperbólico ($z=4$).

De las siguientes ilustraciones se puede volver a comentar el error reproducido a través de una función de distribución acumulativa. Para el caso de los puntos del plano ' $z=2$ ', que es lo que representa la ilustración siguiente, se observa que con este método se disminuye mucho el error de posicionamiento cometido en métodos anteriores (modo independiente); lo que implica que, al posicionar con todas las balizas al mismo tiempo, se logra corregir la dispersión que se veía con anterioridad al emplear este método. No se puede comparar este caso con los resultados de fusión porque no se ha hecho para los datos hiperbólicos, pero, al igual que para el caso que se acaba de ver con datos esféricos, se puede suponer que el uso de las 15 balizas al unísono mejora de manera considerable los resultados del cálculo hiperbólico que se obtendrían al emplear el método de fusión.

De esta ilustración cabe destacar que, al ser cálculo hiperbólico, hay más diferencia entre el posicionamiento de los distintos puntos del plano debido a la menor precisión del método comparado con el caso esférico anterior. Aun así, se hablan de unas magnitudes de error más sensatas en comparación con el caso del modo independiente.

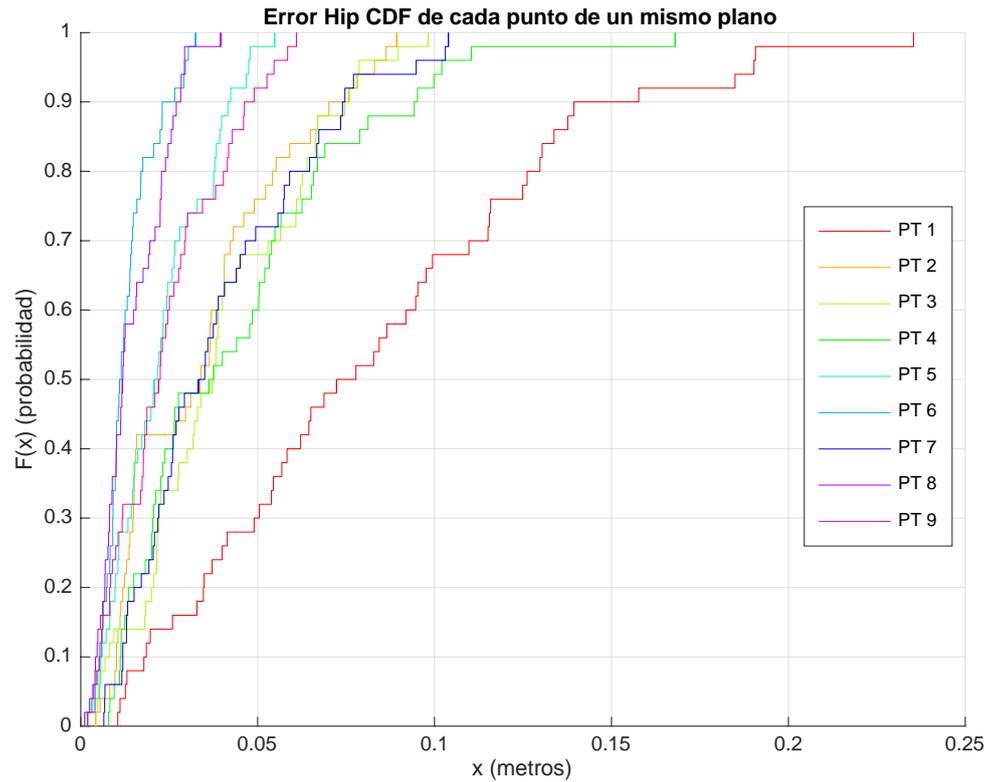


Ilustración 60. Función CDF que representa el error de los 9 puntos que se sitúan en el plano 'z=2' (Gauss-Newton hiperbólico).

La siguiente ilustración 61, aumento de la ilustración anterior, se observa que el 90% de los puntos calculados en el plano se posicionan con un error máximo de 16 cm. El 100% de los puntos, se posicionan con un error máximo de 24 cm; mientras que el mejor de los puntos se posicionan con un error máximo de 3'3 cm.

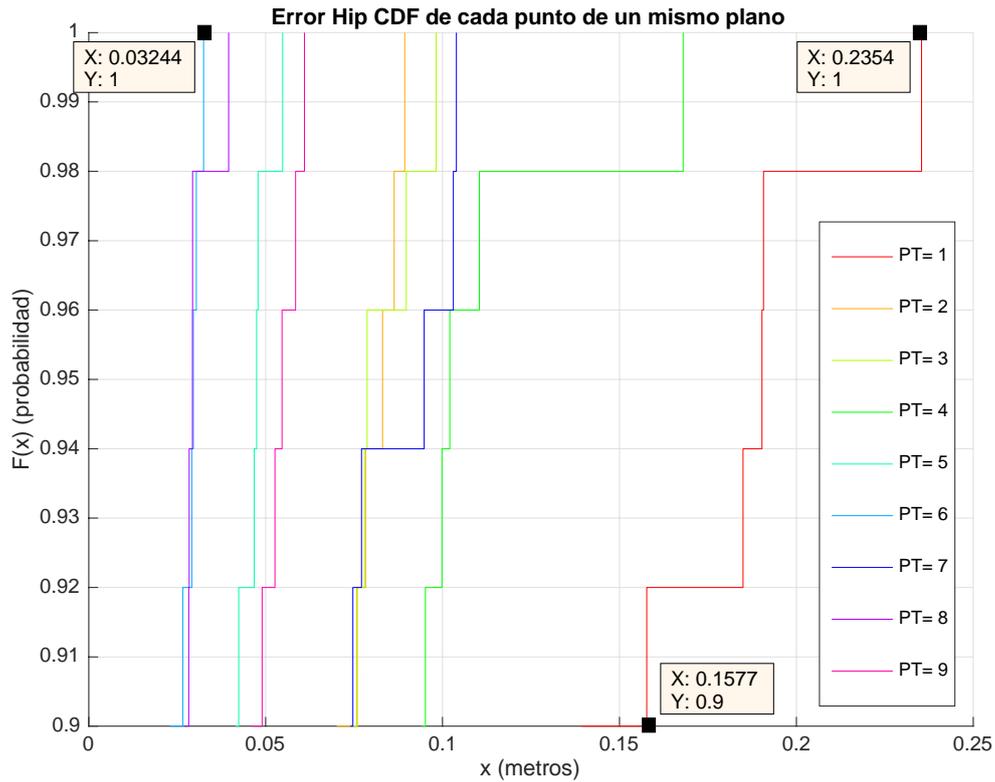


Ilustración 61. Error de posición de los puntos de un mismo plano con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).

En la siguiente ilustración 62 se muestra una comparativa del error de todos los puntos de cada plano que se simulan de manera independiente. De los dos primeros planos cabe mencionar que se posicionan algo mejor los puntos del segundo plano que los del primero, pero con una diferencia de error mínima; mientras que el tercer plano, al tener el problema de la indeterminación del algoritmo para concretar la posición de los puntos, se observa que se tiene un error muy grande a lo largo de todo el plano. Es más, más de un 50% de los puntos de ese plano supera más de la mitad del error cometido a lo largo del plano.

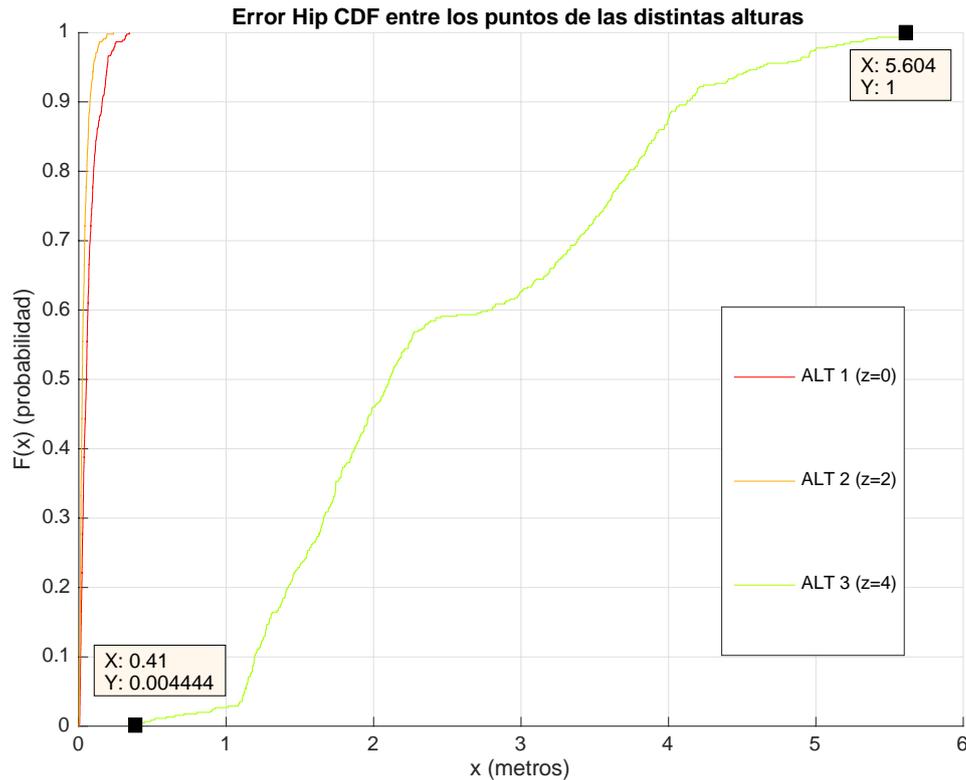


Ilustración 62. Función CDF que representa el error de todos los puntos que se sitúan en los 3 plano diferentes (Gauss-Newton hiperbólico).

La siguiente ilustración 63 es un aumento de la ilustración anterior y se centra únicamente en las curvas de error de los dos primeros planos que es en donde el algoritmo funciona de manera correcta. Se puede concluir de lo visualizado en la ilustración que, el 90% de los puntos de ambos planos se posicionan con un error menor de 16 cm, en comparación con el tercer plano que anda en torno a los 40 cm de error. El 100% de los puntos de ambos planos se posicionan con un error menor de 35 cm, en comparación con el tercer plano que supera los 5 m. Por último, en el plano 'z=2' se logra posicionar el 100% de los puntos con un error máximo de 23'5 cm. Se puede concluir que hay una gran diferencia entre los puntos que se sitúan en un plano que causa indeterminaciones en el algoritmo, y los puntos en aquellos planos donde el algoritmo es capaz de posicionarse.

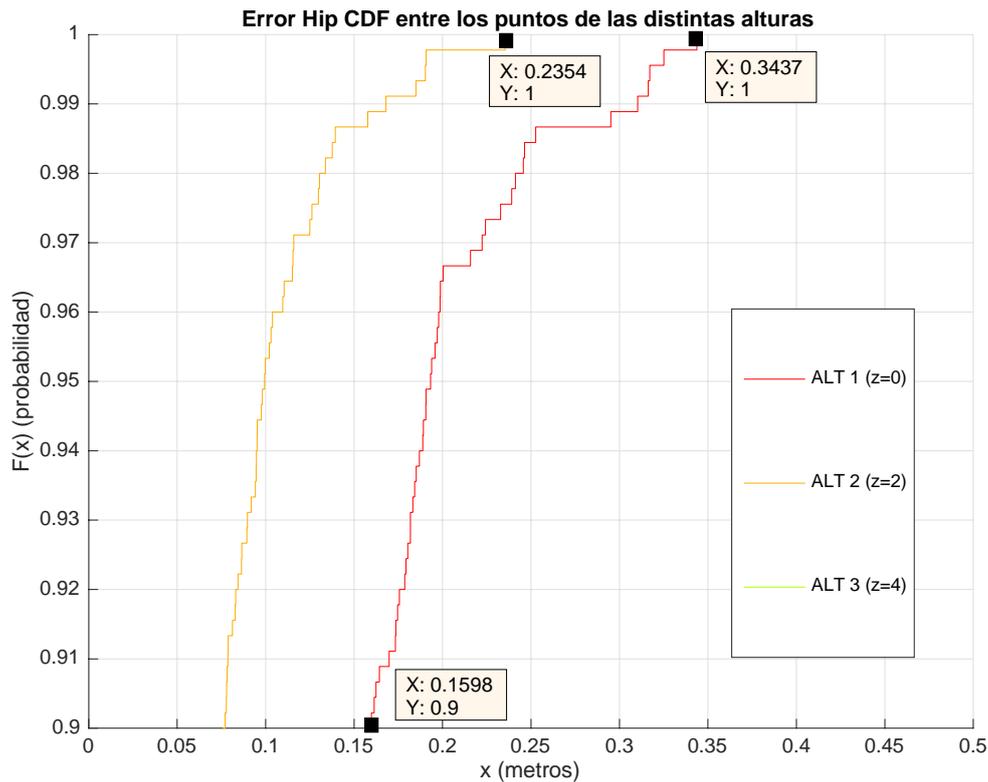


Ilustración 63. Error de posición de todos los puntos de distintos planos con una probabilidad mayor del 90% (Gauss-Newton hiperbólico).

6.3.3 Pérdida de balizas

En las siguientes ilustraciones se plantea un escenario más real que el planteado anteriormente. En los resultados de simulación que se han encontrado hasta ahora a lo largo de todo el documento, se ha planteado una situación bastante ideal puesto que, aunque se haya introducido un error en las medidas del sistema, a recodar, la desviación estándar que representa el ruido implícito del sistema que se aplica en el cálculo de distancias, siempre se consiguen todas las medias que se necesitan de todas las balizas para el cálculo de distancias.

En una situación más realista, puede ser que no siempre se obtenga medidas de todas las balizas debido a obstrucciones por el camino, errores en la propia baliza, errores en la propia medida, fallo de algún sistema, etc. Es por ello, que los resultados que se representan a continuación hacen referencia a esto que se esta comentando.

Se va a empezar analizando los resultados a partir de las posiciones calculadas a través del algoritmo esférico, para posteriormente pasar al algoritmo hiperbólico. Los resultados que se van a exponer únicamente representan el caso de modo simultáneo debido al gran número de balizas que se tienen, y porque ese caso ofrece una visión más orientativa de lo que se quiere mostrar con estas simulaciones.

En esta primera ilustración 64, se muestran los resultados de posicionamiento en el plano 'z=2' con el algoritmo esférico con la pérdida de las medidas correspondientes a cuatro, siete y diez balizas de manera aleatoria. La elección de las balizas a partir de las cuales se pierden sus

medidas se realiza de manera aleatoria para representar ese caso más realista y para que no de la casualidad de representar el mejor caso en el que esa pérdida afecte menos de lo debido.

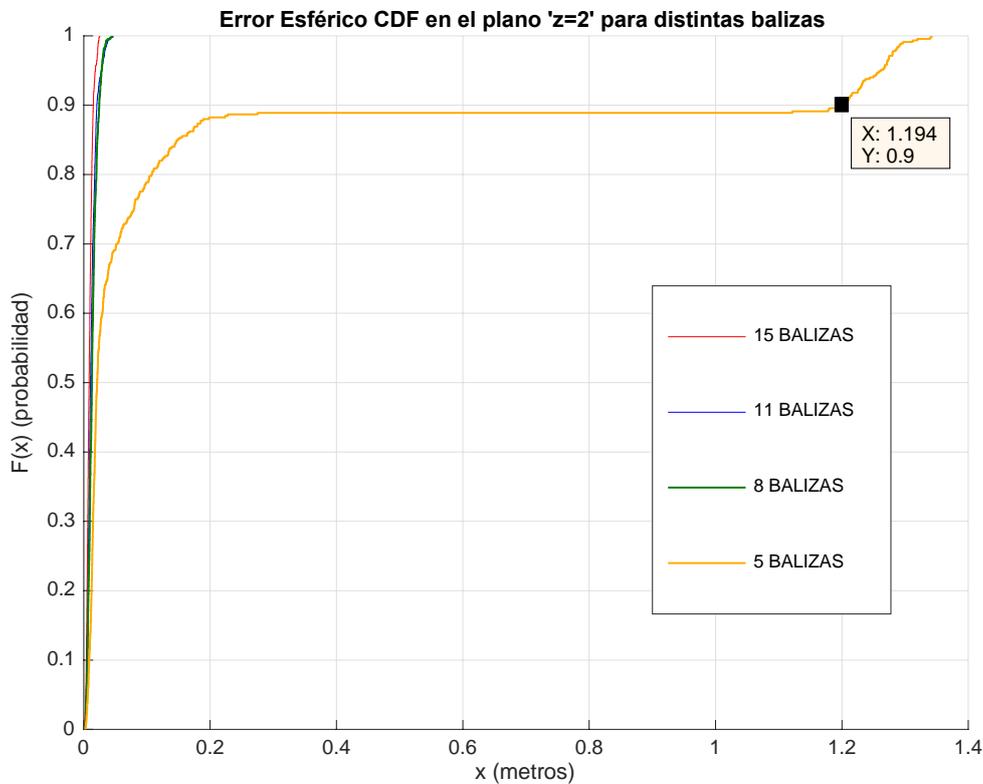


Ilustración 64. Función CDF que representa el error de todos los puntos situados en un mismo plano ($z=2$) para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton esférico).

De la ilustración anterior cabe indicar que se encuentra un punto de inflexión dentro del número de balizas del que se pierden las medidas. Si se pierde entre siete balizas, caso representado en color verde, y diez balizas, caso representado en color amarillo, se observa que se pierde precisión a la hora del cálculo del posicionamiento. Grosso modo se puede observar que el 90% de los puntos se pueden estimar con un error menor de 1,2 metros para el caso de tener únicamente 5 balizas de las que obtener medidas. El punto de inflexión viene dado en torno al 50% de los puntos calculados, puesto que, a partir de ahí, el resto de puntos aumentan en error.

En la siguiente ilustración 65, aumento de la Ilustración 64, se representa lo mismo comentado anteriormente, pero con un poco más de detalle. De la ilustración se puede corroborar al 100% cómo a lo largo de todo el proceso de posicionamiento, los puntos calculados con las mediciones de 5 balizas tienen mucha más desviación de error que en el resto de casos. Se puede corroborar también que se tiene el punto de inflexión comentado en torno al 50% de los puntos posicionados; por lo que en más del 50% de los puntos el error va aumentando de manera exponencial.

También cabe comentar que el 80% de los puntos se calculan con error de 10 cm en los puntos calculados sólo con 5 balizas. Para el resto de casos se logran obtener unos posicionamientos del 80% de los puntos con un error menor de 2 cm.

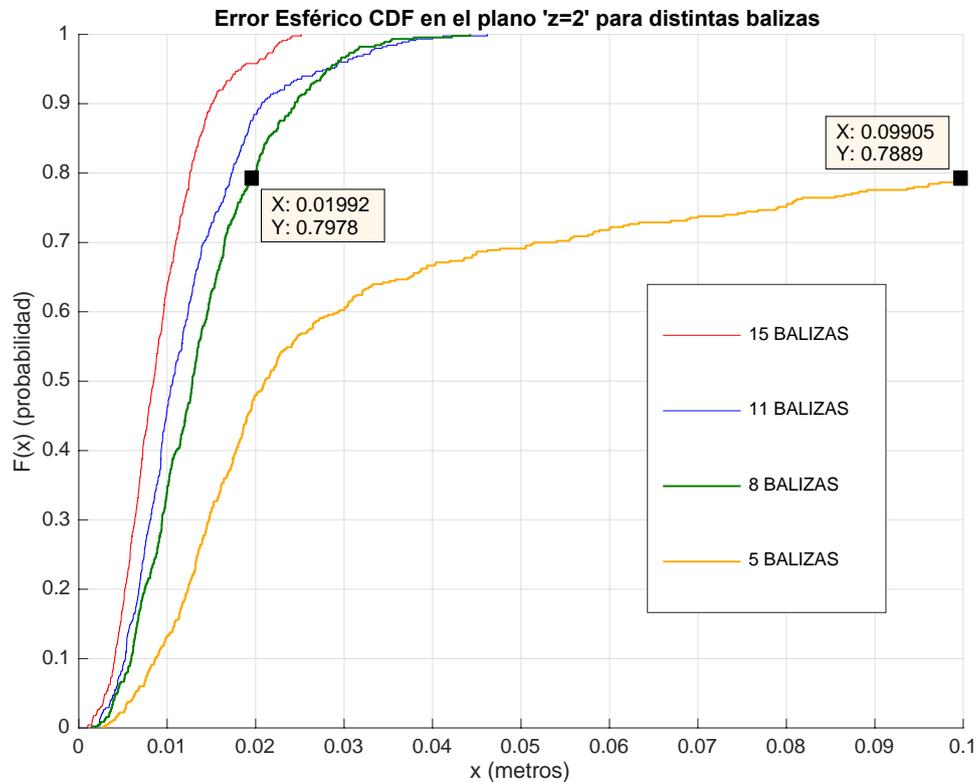


Ilustración 65. Error de posición de 10 cm de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton esférico).

En la siguiente ilustración 66, aumento de las Ilustraciones 64 y 65, se puede visualizar que errores se tienen con el cálculo de posicionamiento a partir de las mediciones obtenidas con 15, 11 y 8 balizas. Se puede ver que para el 100% de los puntos, el error máximo tiene una magnitud de 4,6 cm para el caso de las mediciones con 11 balizas.

Se puede observar también en la ilustración que el mejor caso con mínimo error es el que se representa con las mediciones de las 15 balizas, que es el número máximo de balizas que se tiene en el sistema. Cabe pensar que, si se puede aumentar el número de balizas se obtendrán mejores resultados presuponiendo que se obtendrán mejores posicionamientos con un error menor del actual, pero es mejor trabajar con menos balizas mejor posicionadas. Para este caso de 15 balizas, el error máximo que se obtiene para el 100% de los puntos estimados es menor de 2,5 cm.

Cabe comentar también que para el posicionamiento con las mediciones de 11 y 8 balizas los resultados de error son muy parecidos como se puede apreciar en la ilustración; por lo que la pérdida de esas balizas no supone mucho problema.

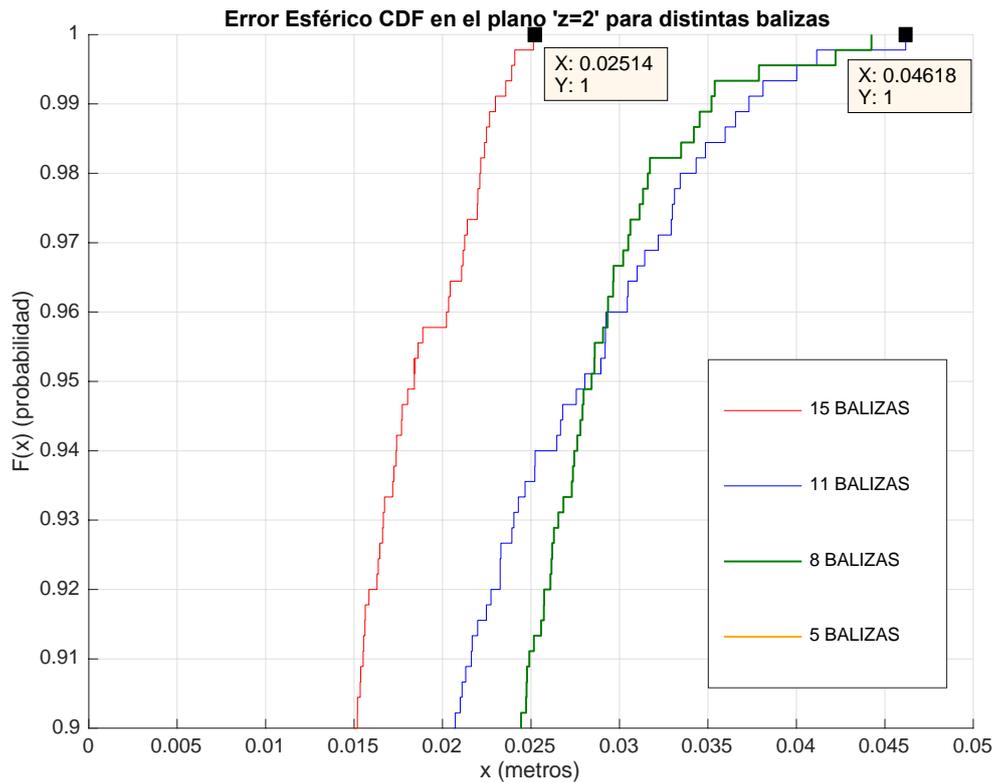


Ilustración 66. Error de posición de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas con una probabilidad mayor del 90% (Gauss-Newton esférico).

A continuación, se muestran los resultados de posicionamiento en el plano ' $z=2$ ' con el algoritmo hiperbólico con la pérdida de las medidas correspondientes a cuatro, siete y diez balizas de manera aleatoria.

De la ilustración 67 siguiente principalmente cabe indicar las distintas magnitudes de error con las que se trata respecto a las ilustraciones anteriores. Esto como siempre se debe a las diferencias entre los algoritmos esféricos e hiperbólicos.

También, se vuelve a encontrar un punto de inflexión que afecta al número de balizas de las que se pierden las medidas. En este caso, este punto de inflexión comienza a priori a partir del 30% de los puntos calculados. Si se pierden entre siete balizas, caso representado en color verde, y diez balizas, caso representado en color amarillo, se observa el error de posicionamiento empieza a aumentar de manera exponencial. Grosso modo se puede observar que el 95% de los puntos se pueden estimar con un error menor de 6 metros para el caso de tener únicamente 5 balizas con las que realizar las medidas.

También cabe destacar que en el 90% de los puntos que se calculan se comete la mitad del error que se tiene en total frente al 10% restante donde parece que el error tiene una pendiente menos pronunciada.

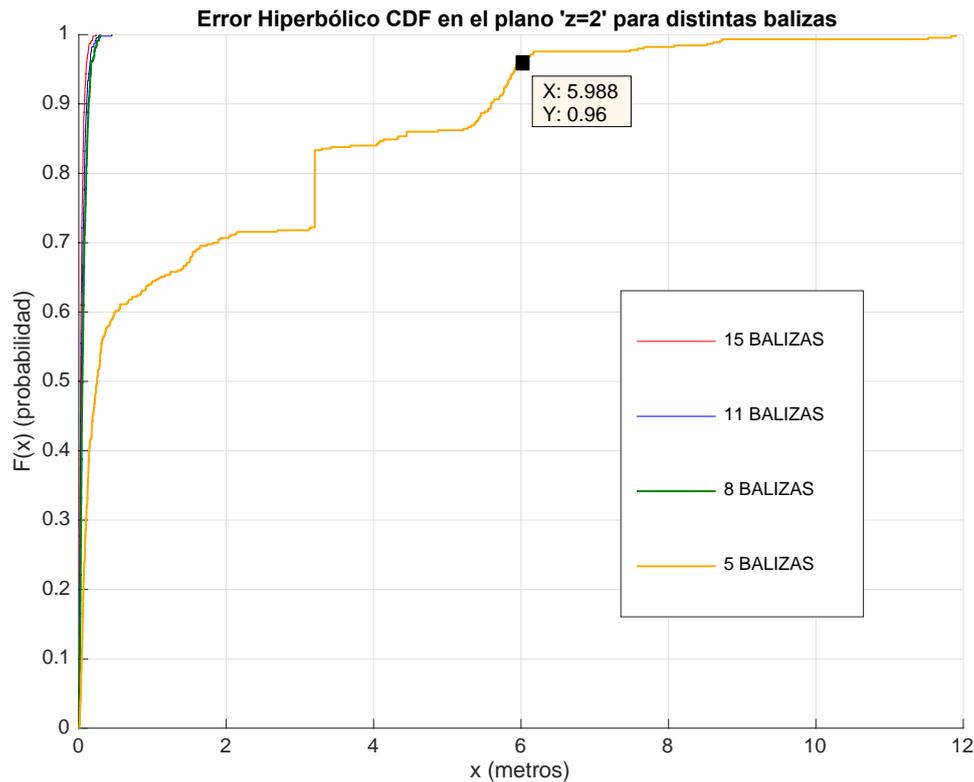


Ilustración 67. Función CDF que representa el error de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton hiperbólico).

En la siguiente ilustración 68, aumento de la Ilustración 67, se representa con más de detalle que el 100% de los puntos calculados con las mediciones de 5 balizas tienen mucha más desviación de error con respecto al resto de casos.

También se corrobora mejor que en este caso directamente no se puede hablar de un punto de inflexión como tal debido a que desde el inicio se visualiza que el error de posicionamiento no sigue la misma línea que el resto de casos. A partir del 40% de los puntos calculados, se empieza a disparar el error de manera acuciante.

También cabe comentar que el 60% de los puntos se calculan con error de 0,5 m en los puntos calculados sólo con 5 balizas. Para el resto de casos se logran obtener unos posicionamientos del 80% de los puntos con un error menor de 10 cm.

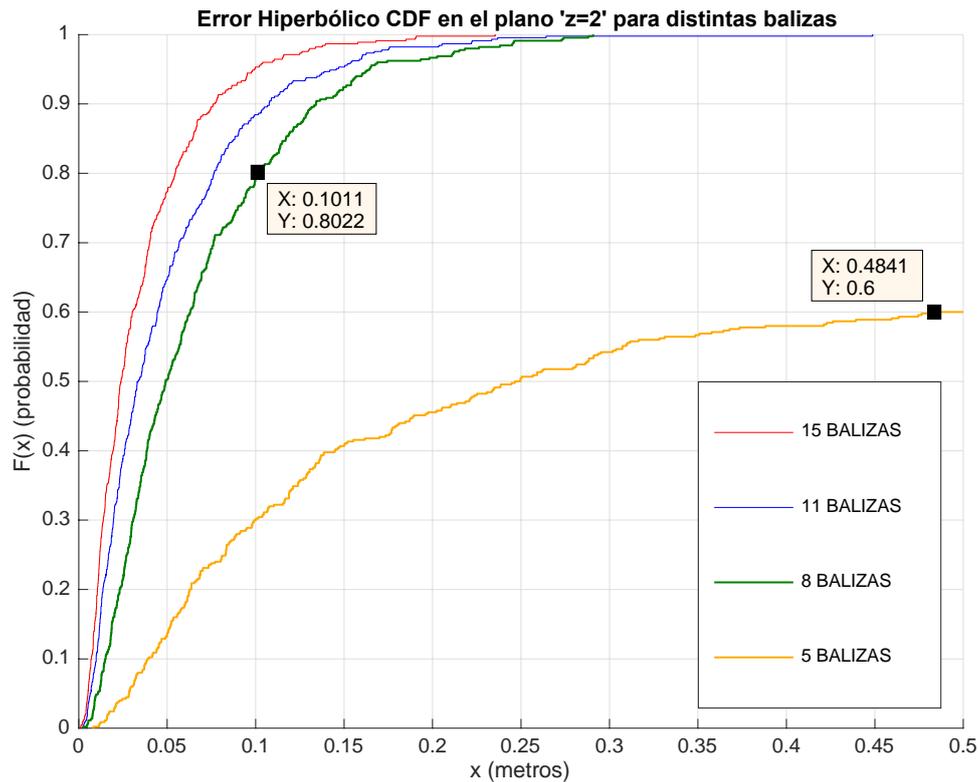


Ilustración 68. Error de posición de 0,5 m de todos los puntos situados en un mismo plano ($z=2$) para el caso de pérdida de mediciones de distintas balizas (Gauss-Newton hiperbólico)

En la siguiente ilustración 69, aumento de las Ilustraciones 67 y 68, se puede visualizar qué errores se tienen con el cálculo de posicionamiento a partir de las mediciones realizadas con 15, 11 y 8 balizas. Se puede ver que para el 100% de los puntos, el error máximo tiene una magnitud de 45 cm para el caso de las mediciones con 11 balizas.

Se puede observar también en la ilustración 69 que el mejor caso con mínimo error es el que se representa con las mediciones de las 15 balizas, puesto que es el número máximo de balizas que se tiene en el sistema. Cabe destacar que en los tres casos que se representan, no hay mucha diferencia entre unos y otros debido a que la diferencia entre casos en el 90% de puntos calculados es de 5cm; por lo que, para este caso, con 8 balizas se obtendría un posicionamiento bastante bueno sin necesidad de tener más balizas; aunque obviamente cuantas más balizas se tengan, más error se irá disminuyendo.

Para el caso óptimo que es el de las 15 balizas, el 100% de los puntos estimados se calculan con un error máximo de 23,5 cm.

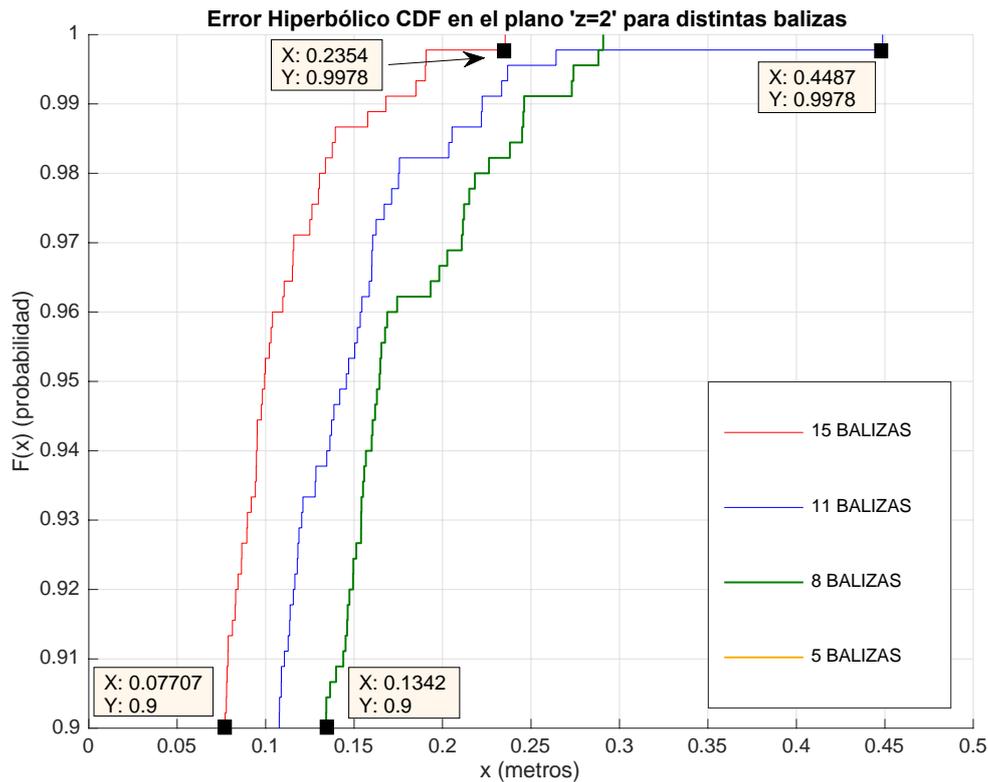


Ilustración 69. Error de posición de todos los puntos situados en un mismo plano para el caso de pérdida de mediciones de distintas balizas con una probabilidad mayor del 90% (Gauss-Newton hiperbólico)

6.3.4 PDOP de modo simultáneo

Después de analizar los resultados obtenidos a partir del empleo del modo simultáneo, se procederá a calcular el PDOP para ver el comportamiento que tienen esos datos dentro del escenario establecido con el uso al algoritmo ya visto.

Dentro del estudio del PDOP se analizarán los datos del algoritmo esférico e hiperbólico a través del cálculo del PDOP mediante métodos heurísticos y mediante la inversa de la matriz Jacobiana.

6.3.4.1 PDOP mediante métodos heurísticos

Para comenzar, se empezarán analizando los datos esféricos mediante el algoritmo de métodos heurísticos. Estos datos son los que se encuentran en la siguiente ilustración 70.

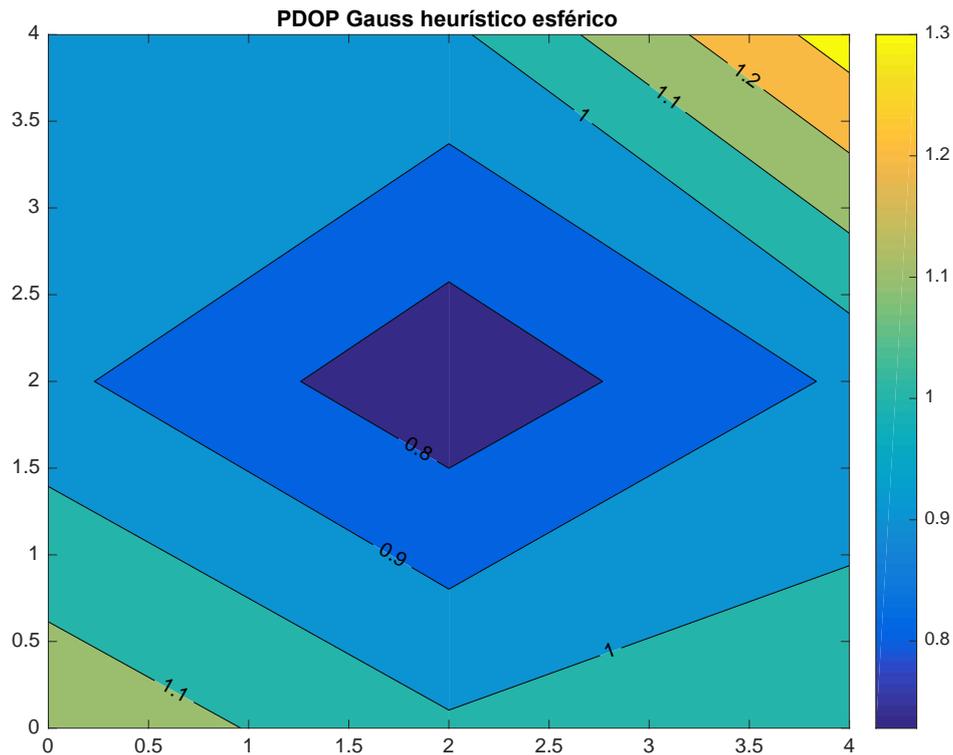


Ilustración 70. PDOP del posicionamiento esférico mediante métodos heurísticos ($z=2$).

Comparando la ilustración anterior con la Ilustración 32 y la Ilustración 46, donde se comparan mismos métodos de aplicación de PDOP sobre los distintos modos de obtención de resultados; se puede observar que en esta última ilustración la precisión de posicionamiento es mejor puesto que se minimiza el error. Esto es debido a la utilización de varios sistemas de balizas a la vez para calcular una misma posición. Calculando una posición a partir de los 3 sistemas de balizas de manera simultánea, se logran que los resultados del PDOP sean uniformes respecto a la baliza central de todos los sistemas de referencia.

En la siguiente ilustración, se muestran los resultados de PDOP mediante el algoritmo hiperbólico usando métodos heurísticos. Comparando la ilustración siguiente con la Ilustración 33, se vuelve a comprobar que debido al uso de los 3 sistemas de balizas se reduce el error en un 95%; por lo que se obtiene para estos resultados una buena precisión. Debido al método hiperbólico en sí mismo, las posiciones de las cuales se obtienen los mejores resultados son las que se encuentran más cerca a los 3 sistemas de balizas. Estas son las posiciones que se encuentran en la esquina superior derecha.

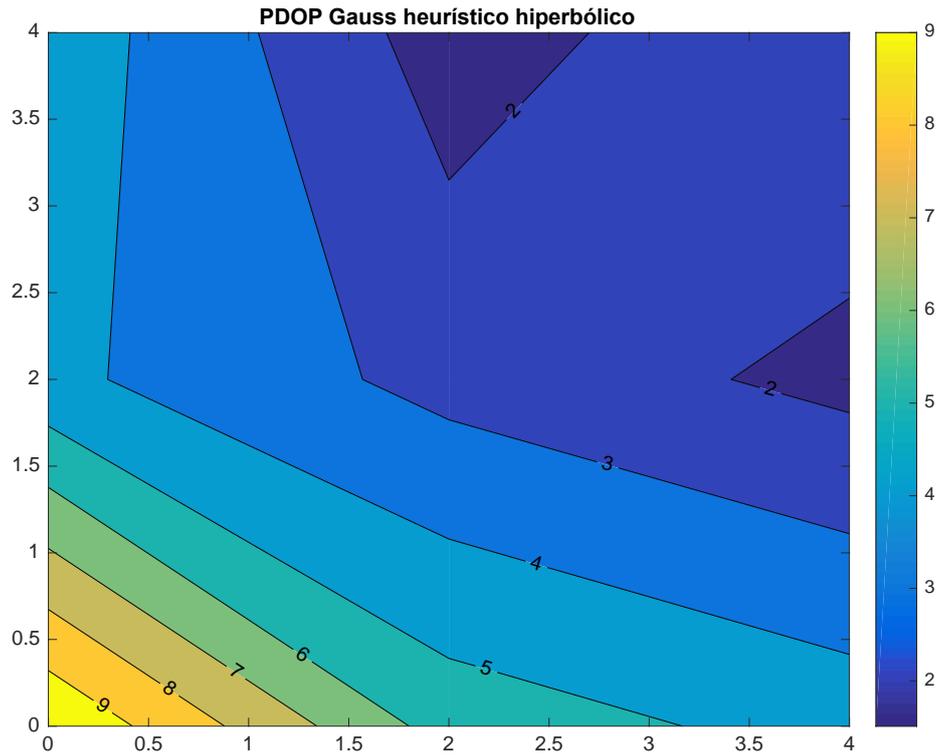


Ilustración 71. PDOP del posicionamiento hiperbólico mediante métodos heurísticos ($z=2$).

6.3.4.2 PDOP mediante la inversa de la matriz Jacobiana

En las ilustraciones siguientes se aplica el método de la función inversa para ambas trilateraciones. En la Ilustración 72, se muestran los resultados esféricos de PDOP. Si se comparan estos datos con los presentados en la Ilustración 34, se vuelve a comprobar que se reduce el error debido a los 3 sistemas de balizas empleados y la precisión se representa de manera uniforme alrededor de donde se sitúa la baliza central de todos los sistemas de referencia.

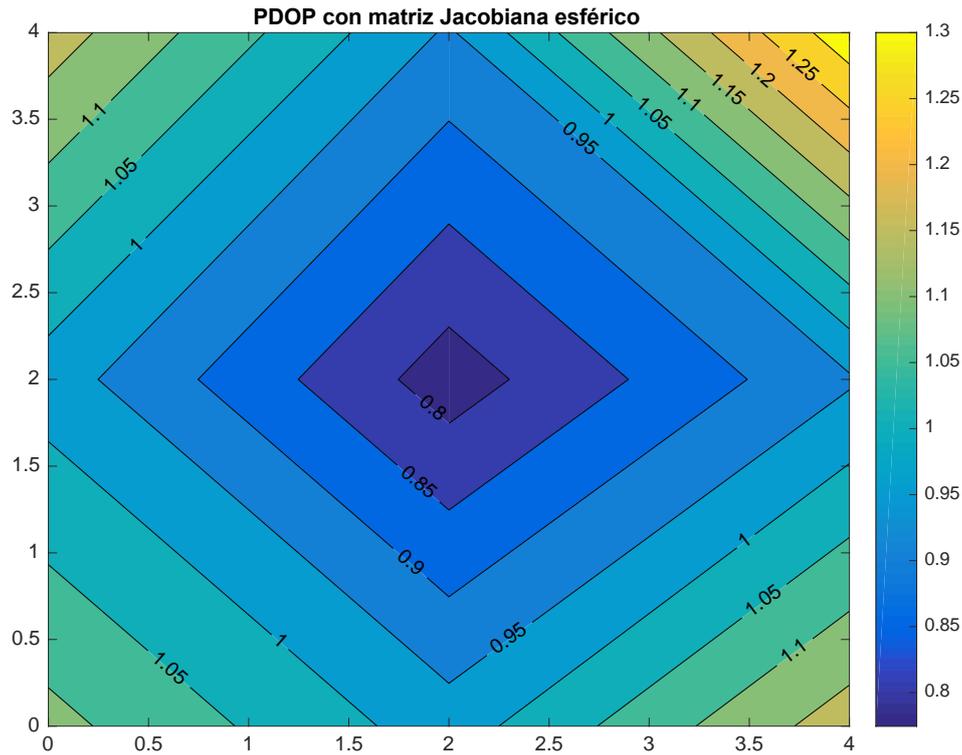


Ilustración 72. PDOP del posicionamiento esférico mediante la inversa de la matriz Jacobiana ($z=2$).

En la ilustración 73 siguiente, se muestran los resultados hiperbólicos de PDOP mediante la aplicación de la función inversa. Si se comparan estos datos de PDOP con los presentados en la Ilustración 35, se vuelve a comprobar que se reduce el error en torno al 95% debido a los 3 sistemas de balizas empleados. La precisión de los datos vuelve a ser mejor en la esquina superior derecha debido a la posición de esos sistemas de referencia. Esto es debido a que el algoritmo hiperbólico emplea distancias hiperbólicas basadas en la diferencia entre distancia de balizas. Es por esto que al estar más cerca de los sistemas de referencia la precisión es mejor. Aun así, en la esquina opuesta los resultados son totalmente válidos puesto que la precisión del posicionamiento en esos puntos es bastante menor que la del resto de cálculos de la posición.

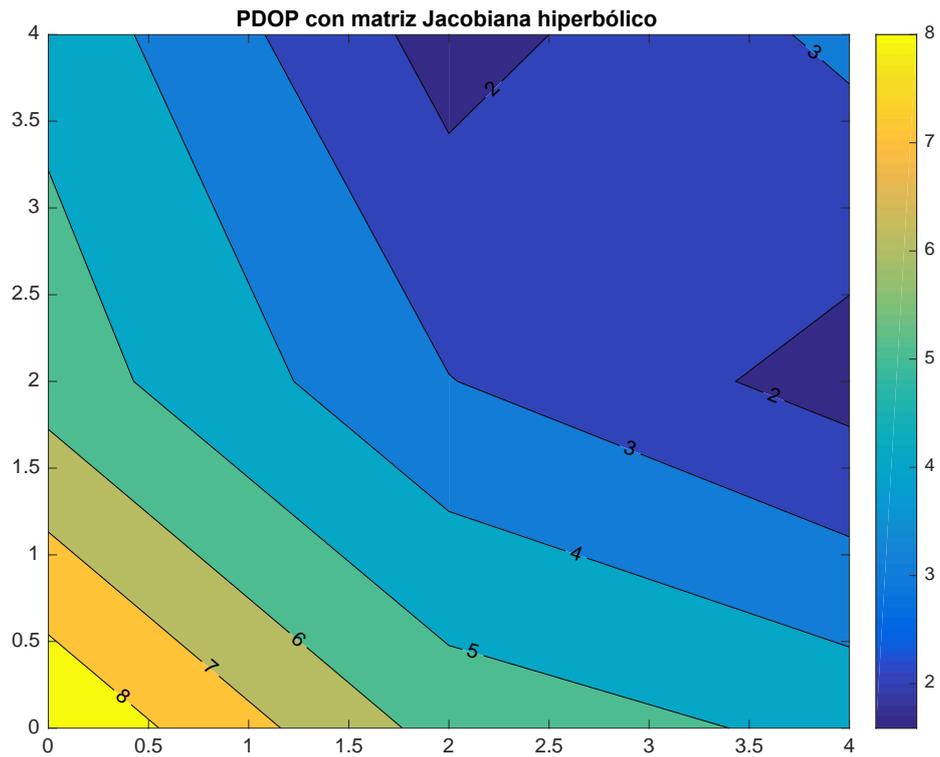


Ilustración 73. PDOP del posicionamiento hiperbólico mediante la inversa de la matriz Jacobiana ($z=2$).

7 CONCLUSIONES

Para concluir con los análisis de los resultados, se puede acabar con las siguientes conclusiones:

- Se elige el método de Gauss-Newton como el método para la representación de todos los resultados porque con él se posicionan mejor los puntos a estimar y es ampliamente usado en la literatura. No obstante, en algún caso hay problemas con la inicialización del algoritmo.
- En cuanto a la representación de resultados, se puede concluir que es más preciso el posicionamiento mediante el método esférico que el hiperbólico.
- En cuanto a la elección del método de fusión de datos, se ha elegido el método clásico de estimación por máxima verosimilitud (MLE) debido a que pondera cada dato según su varianza. Es una primera aproximación de cómo sería la fusión de datos. Un estudio posterior que surgiría como continuación de éste sería el desarrollo de otros métodos de fusión mediante el uso de otras técnicas (sobre todo para no tener que generar datos estadísticos a priori en las distintas posiciones).
- Con el algoritmo de fusión se consigue mejorar significativamente las prestaciones del sistema en cuanto a precisión, debido a que se corrigen los errores típicos de la coordenada perpendicular del sistema de balizamiento en el caso 3D.
- Se puede verificar que entre todos los modos de simulación que se han empleado para la obtención de los distintos datos, cuantas más balizas se pongan situadas en distintos planos, se obtendrá una mayor precisión de estos reduciendo la dispersión; pero es crucial la posición de las balizas en relación al entorno de posicionamiento.
- Debido a la conclusión anterior, se verifica que la obtención de un posicionamiento en un punto dado es mejor cuando se realizan las medidas de manera simultánea a través de todas las balizas de las que se dispongan. Los métodos de fusión de datos dan buenos resultados, pero no llegarán a ser tan precisos como el cálculo de todo el conjunto de manera simultánea.
- Por otro lado, hay que tener en cuenta que cuando se consideran todas las balizas simultáneamente, la pérdida de los datos proporcionados por las balizas no penaliza en exceso la precisión de la localización, siempre y cuando la pérdida de medidas no supere el 50 %.
- A las conclusiones anteriores también se puede añadir, viendo todos los resultados, que es muy importante donde se sitúen los sistemas de balizas. Se deja para futuros estudios el contemplar cual sería el mejor punto donde posicionar esas balizas, y el mínimo número de ellas, para que se pudieran mejorar los errores de sistema.
- También cabe destacar que entre todos los métodos de estimación del PDOP, es más fiable y preciso el método de la obtención de esta medida a través del uso de la función inversa, donde se obtiene el PDOP a través de la aplicación inversa del algoritmo de posicionamiento (pero como el heurístico, ya depende del sistema de posicionamiento en sí y no solo de consideraciones geométricas).
- La representación del error a través de una función acumulativa, como puede ser la función CDF usada en este trabajo, muestra una buena representación del error que indica la variación del posicionamiento de los puntos en los casos indicados.

Cabe destacar, que los resultados comentados en ese trabajo son el resultado de la aplicación de unos algoritmos de posicionamiento en un entorno determinado. No se han tenido en cuenta las circunstancias que se tendrían en un sistema real como podrían ser: la sincronización entre

todas las balizas para poder obtener una posición mediante el modo simultáneo, donde se encontraría el límite de cobertura con el que trabajarían las balizas, los sensores encontrados en ellas para ver dentro de que límite real se permite el posicionamiento; etc.

Todo ello implica que, aunque haya métodos con los que se pueda posicionar un móvil con más error que otros, quizás se coloquen en las primeras posiciones de las listas en ser los más usados los que tengan más error debido a que implican menores problemas prácticos. Sólo hay que hacer balanza entre lo que más conviene en cada momento, para ver si se está dispuesto a sacrificar un poco de precisión a la hora del posicionamiento por menos problemas a la hora del montaje y uso del sistema LPS.

VII. DIAGRAMA SW

Para organizar un poco las ideas de todo lo que se acaba de analizar en los anteriores apartados, se van a presentar un conjunto de diagramas SW para clarificar las acciones llevadas a cabo en la programación de los algoritmos.

El primer diagrama que se muestra a continuación señala de manera genérica todos los scripts que se han programado para el desarrollo de este documento. El script 'lps_unico.m' representa los resultados de posicionamiento con un único grupo de balizas, mientras que el script 'lps_fusion.m' representa los resultados procedentes de la fusión de los datos cuando se intenta ampliar a un sistema de 3 conjuntos de balizas que trabajan en modo independiente. El último script que se emplea para obtención de resultados de posicionamiento es el de 'lps_triple.m'. En este script se obtiene el posicionamiento a través de los 3 conjuntos de balizas utilizados de manera simultánea.

Por otro lado, el resto de scripts representan los cálculos del PDOP correspondientes a cada uno de los métodos desarrollados en los anteriores scripts ya comentados.

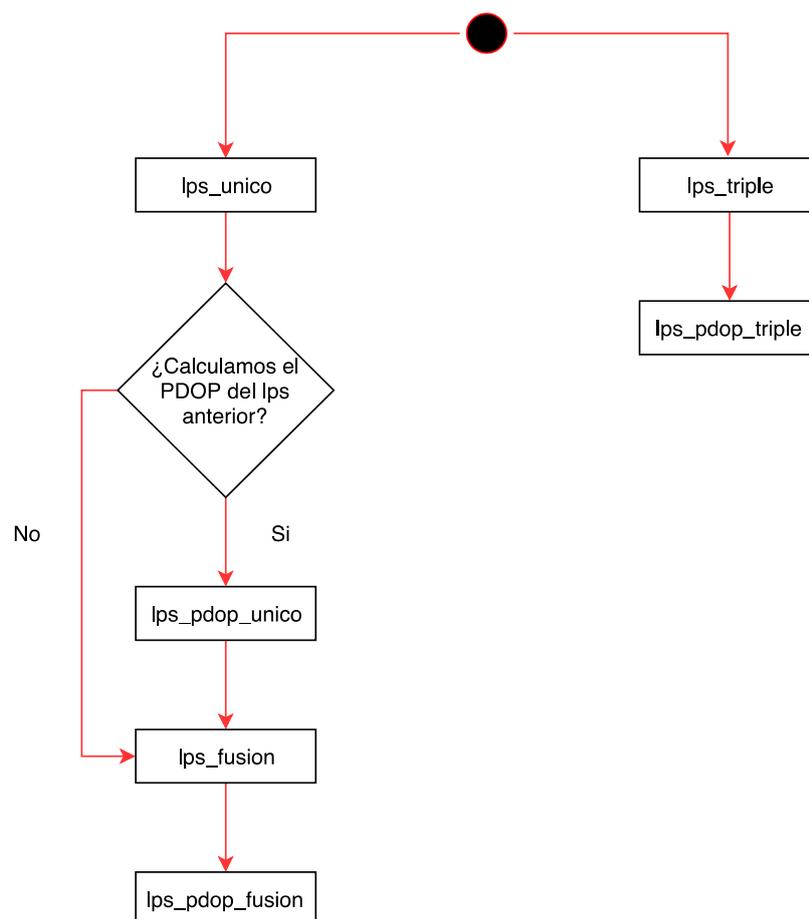


Ilustración 74. Diagrama de SW genérico.

La siguiente ilustración muestra el diagrama de SW para tener una idea del algoritmo empleado para los scripts de los sistemas de un solo conjunto de balizas o de los 3 conjuntos de balizas.



Ilustración 75. Diagrama SW de los scripts referentes a un único conjunto de balizas y a 3 sistemas de balizas.

Cabe destacar de este diagrama que el cálculo del posicionamiento se realiza tanto para el algoritmo esférico como el hiperbólico. Es por ello que el cálculo de la distancia se hace del mismo modo; de manera directa entre el punto a posicionar y las balizas, o mediante el cálculo hiperbólico de la distancia. El cálculo del error representa el cálculo de todos los tipos de errores

comentados, y la celda de inclusión de ruido representa el ruido que se introduce en el sistema de manera implícita.

El siguiente diagrama muestra el desarrollo del script 'lps_fusion.m' que, si se recuerda, necesita de los valores calculados con anterioridad en el script 'lps_unico.m'.

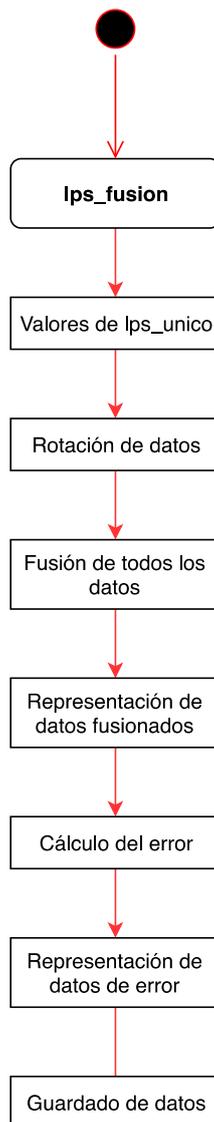


Ilustración 76. Diagrama SW del script que intenta replicar el posicionamiento de 3 conjuntos de balizas a través de la fusión de datos.

En el último diagrama que se muestra a continuación, se pueden observar las distintas funciones que aplican para el cálculo del PDOP para cada uno de los casos planteados para la obtención de resultados.

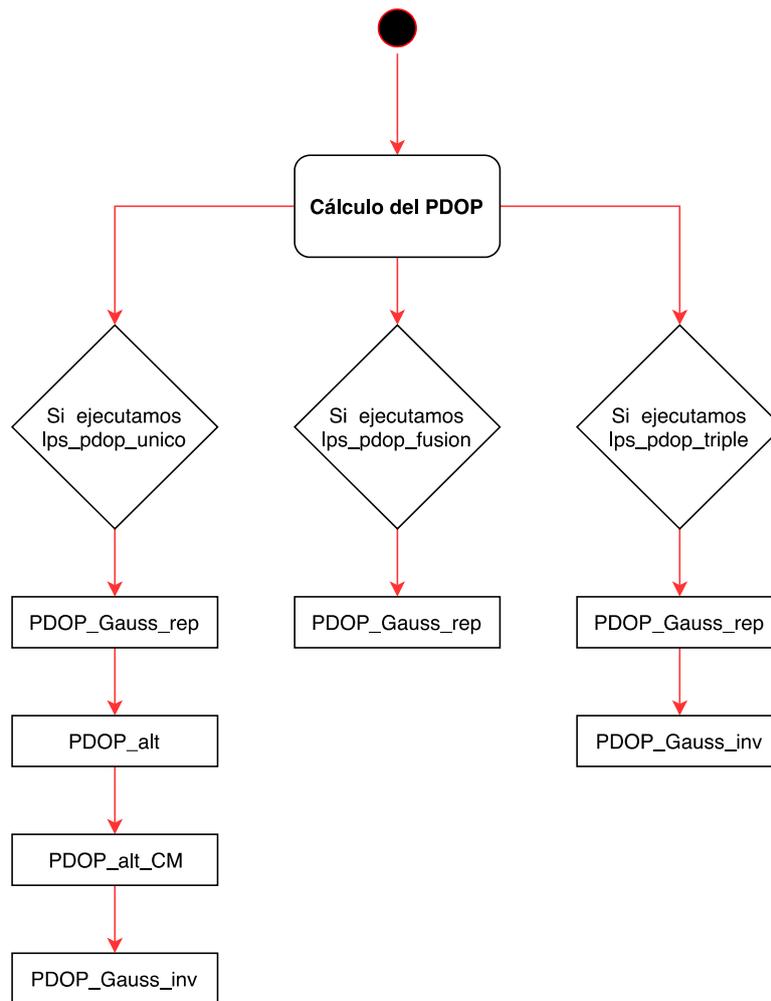


Ilustración 77. Diagrama SW para los scripts de cálculo del PDOP.

Como aclaratorio de las funciones que se utilizan para el cálculo del PDOP, se tiene el cálculo por medio del método heurístico del algoritmo de Gauss-Newton (PDOP_Gauss_rep.m), y mediante el cálculo de la inversa de la matriz Jacobiana (PDOP_Gauss_inv.m), también de Gauss-Newton. Ambas funciones se desarrollan con el algoritmo de posicionamiento esférico e hiperbólico.

Por otro lado, se tiene el cálculo del PDOP por medio de la altura. En la función 'PDOP_alt.m' se desarrolla el PDOP a través del cálculo de la altura de todos los posibles tetraedros que se pueden formar con el conjunto de balizas que se utilizan; mientras que con la función 'PDOP_alt_CM.m' se emplea la técnica del Bideterminante de Cayley-Menger para el cálculo de esas alturas. Para más detalles, se remite al punto 3 donde se realiza un análisis más exhaustivo del cálculo del PDOP.

VIII. PLIEGO DE CONDICIONES

Para la realización de este trabajo, se ha requerido de una serie de materiales con ciertas restricciones para poder cumplir con lo obtenido a lo largo de este documento.

Los requisitos básicos para poder realizar este proyecto son:

- Un ordenador.
- Matlab.
- Conexión a Internet.
- Acceso a artículos académicos.

El ordenador está sujeto a una serie de condiciones en cuanto a potencia de computación. La ejecución del software requiere de un ordenador que sea capaz de ejecutar todos los cálculos complejos que tienen los algoritmos de una manera rápida y eficiente. En el caso de que el ordenador que se tenga no cumpla con esas condiciones, hay otros métodos que se pueden emplear para la ejecución de los scripts desarrollados. Uno de esos métodos puede ser el uso de servidores para la realización de esas operaciones matemáticas.

El software de Matlab únicamente requiere de una versión bastante actualizada que tenga las funciones que se emplean en los algoritmos. Para este trabajo se ha empleado una versión del 2015 y no se ha generado durante la duración del trabajo ningún problema, pero si es una versión posterior, habrá que asegurar que se está actualizado y que no habrá ningún problema.

Se requiere de conexión a Internet para la búsqueda de distintas consultas y documentación de Matlab para todo tipo de dudas que puedan surgir, y, sobre todo, para tener acceso a los artículos académicos que ayudarán a comprender de una mejor manera toda la matemática interna que conllevan los distintos algoritmos desarrollados y que se han explicado en apartados anteriores de este documento.

Lo comentado anteriormente es lo que se necesita para el desarrollo teórico con su puesta en práctica a nivel de simulación. Para llevar este proyecto a la práctica real, se necesita del conjunto de balizas que se ha utilizado para las simulaciones. Este sistema de balizas es el que se representa en la Ilustración 9. Por otro lado, el conjunto de balizas necesita de una serie de sensores donde se tienen que recoger las medidas de posición del objeto a posicionar. Debido a ello, se requiere de un móvil, un robot mismamente, que se vaya moviendo por el entorno con un sensor o varios desde el que comunicar su posición a las balizas. También habría que marcar unos límites por el que el robot se puede mover para que las balizas puedan captar su señal. Esto dependerá de las características que tengan los sensores de las balizas en cuestión de cobertura.

IX.PRESUPUESTO

A continuación, se van a mostrar los costes generados como resultado de la realización de este proyecto. Esto es indicativo para dar una idea de lo que se necesitaría a la hora de querer replicar estos datos obtenidos anteriormente, o para seguir con futuros desarrollos.

Estos costes tendrán en cuenta tanto la mano de obra utilizada como las necesidades de materiales asociados al desarrollo SW, licencias utilizadas para ese mismo desarrollo y gastos de consumos.

Dentro de estos costes, se empezará mencionando los costes directos. En este caso, los costes directos se refieren a todos los gastos que tiene en una empresa debidos al uso directo, para producir el servicio que lleve a cabo. El único coste directo a tener en cuenta para el desarrollo de este proyecto hace referencia a los desarrolladores que han hecho efectivo este documento; por lo que hay que incluir al autor de este proyecto.

Por ello, la siguiente tabla muestra las horas dedicadas tanto para el desarrollo software que se ha requerido para obtener los resultados expuestos, como al trabajo realizado para la creación de esta memoria donde se comenta todo el proceso del proyecto.

Tabla 1. Costes directos.

Costes directos:	Periodo (Meses)	Tiempo (Horas/Mes)	Coste (€/hora)	(€)
Ingeniero	8	60	85	40800

Por otro lado, hay que añadir a los costes anteriores los costes indirectos que hacen referencia a todos aquellos gastos que tiene una empresa, debidos al uso de todo aquello que se emplea para el desarrollo de un proyecto pero que no influye de manera directa en la producción del servicio que se ofrece. Debido a ello, hay un gran número de servicios y materiales que pasan desapercibidos, pero que, sin ellos, el proyecto no podría haber salido adelante.

Entre esos costes hay que incluir, el ordenador con el que se han desarrollado las simulaciones y la memoria presente, las licencias de Matlab y de Office que han sido claves para el desarrollo del desarrollo SW y el documento escrito respectivamente, la conexión a Internet necesaria para todas aquellas consultas surgidas a lo largo de todo el proceso de desarrollo, los gastos de luz asociados al uso de la herramienta principal que es el ordenador, y por último, los gastos debidos al desplazamiento para el apoyo y la dirección del tutor asociado a este proyecto. Todo lo comentado se resume en la siguiente tabla con sus gastos correspondientes.

Tabla 2. Costes indirectos.

Costes indirectos:	Periodo de amortización (Años)	Coste amortización (€/Año)	Periodo de uso (Año)	Precio (€)	(€)
PC	3	500,00	0,67	1500,00	1166,67
Licencia Matlab	1	800,00	0,67	800,00	266,67
Licencia Office	1	105,60	0,67	105,60	35,20
Conexión Internet	0	0,00	0,67	384,00	384,00
Desplazamientos	0	0,00	0,67	100,80	100,80
Luz	0	0,00	0,67	400,00	400,00

Para finalizar, la tabla siguiente muestra el conjunto de todos los gastos que se deben al desarrollo completo del proyecto. Este resultado ronda los 43.000 € como se puede ver a continuación.

Tabla 3. Costes del proyecto.

Costes totales:	(€)
Costes directos	40800,00
Costes indirectos	2353,33
	43153,33

X.MANUAL DE USUARIO

En este apartado, se procederá a relatar el procedimiento a seguir para que se pueda ejecutar el software desarrollado del que se ha hablado en apartados anteriores.

En la siguiente ilustración 78 se muestra lo que se va a comentar.

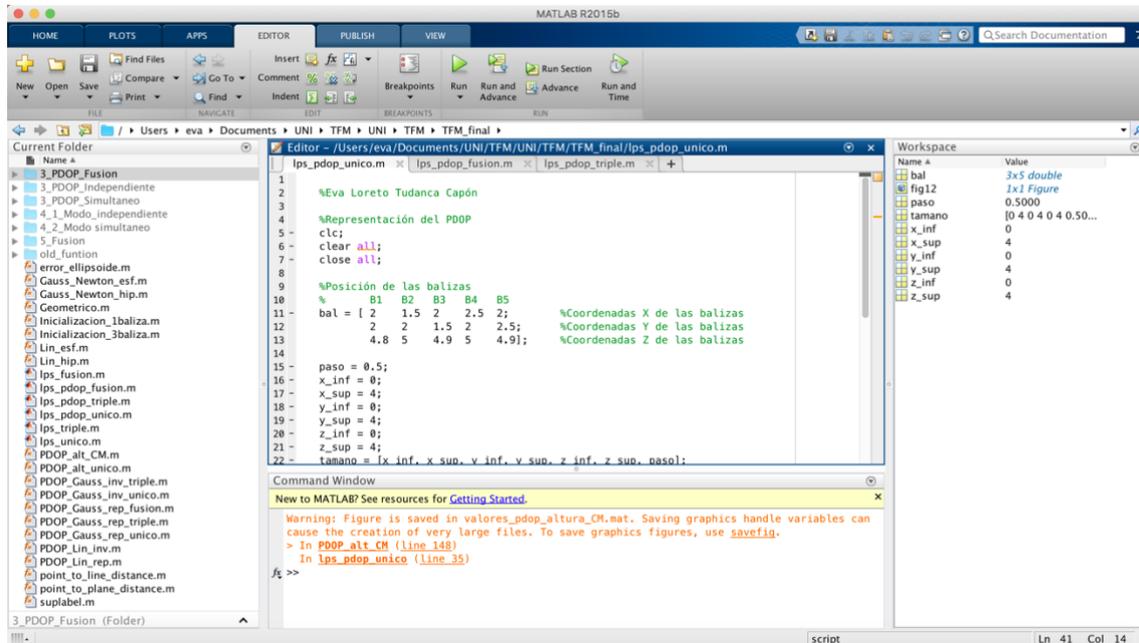


Ilustración 78. Directorio de programación.

En la ilustración anterior se muestra el directorio de programación donde se muestran todas las funciones y script utilizados para el desarrollo de este proyecto.

Como ya se conoce del Matlab, en la parte de la derecha se muestran todas las variables resultantes del desarrollo del script mostrado en la parte central de la pantalla. En la parte inferior se muestra la pantalla de comandos donde se indica el script que se está ejecutando y alguna variable que se puede mostrar por pantalla. En este caso se muestra una advertencia resultante del pintado de los resultados que se obtienen en el script mostrado.

En la parte de la izquierda se muestra una ventana que representa el directorio donde se guardan todos los archivos desarrollados para el proyecto. A continuación, se mostrará una ilustración más ampliada de la anterior para poder observar lo acabado de comentar.

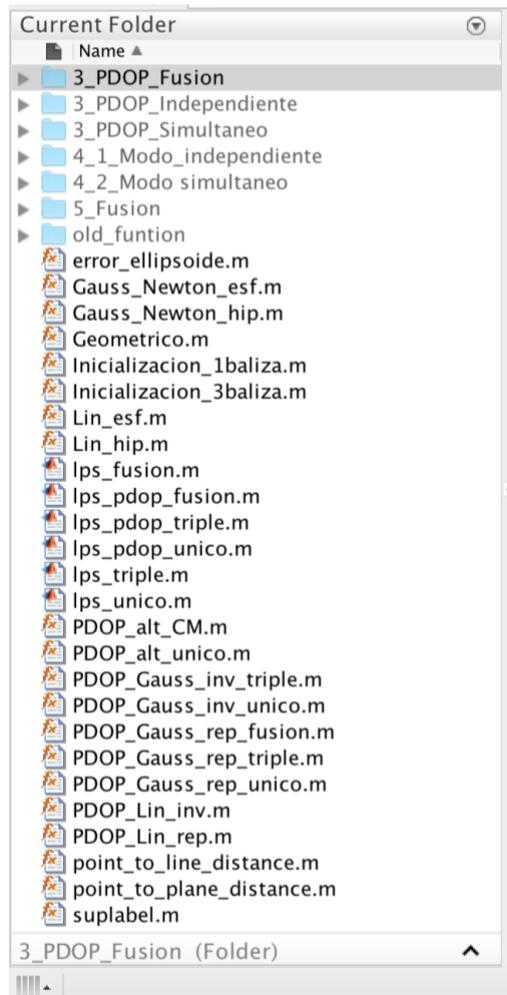


Ilustración 79. Ventana del directorio.

Como se puede visualizar en la anterior ilustración, se encuentran todos los scripts que se han comentado para los diagramas de software, más el resto de funciones necesarias para que la ejecución de los scripts no falle. Las carpetas que se encuentran en este directorio son para clasificar los datos de los distintos algoritmos.

Todas las funciones que se necesitan se encuentran en una misma carpeta por requisito del programa Matlab. Esa carpeta es lo que se conoce como path, y es lo que se puede observar en la pestaña que se encuentra encima de la ventana del directorio y de la pantalla principal donde se muestran los scripts. El path en este caso es el que se puede visualizar en la siguiente ilustración.

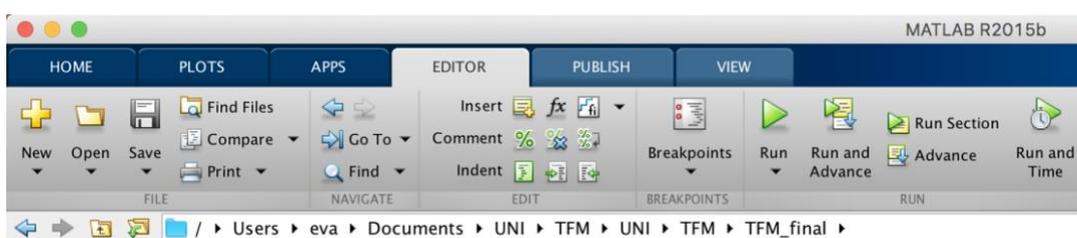


Ilustración 80. Path para el directorio de programación.

Todo lo acabado de comentar hace referencia al programa de Matlab una vez que ya se tiene este instalado. En cuanto al manual de instalación para poder instalar el programa, es bien sencillo y ya conocido puesto que es un programa que se utiliza con frecuencia en la propia docencia de la UAH. Si no se ha trabajado anteriormente con este programa en cuestión, al descargar el programa suele venir un fichero de explicación de instalación, y si no, en internet abundan manuales de instalación para este programa.

En cuanto al mantenimiento del programa ya se ha comentado que lo único que se necesita es tener una versión de software actualizada; y si se necesita cualquier tipo de ayuda con el programa, hay también muchos manuales de usuario por Internet y páginas de consulta para cualquier ayuda sobre la programación.

Las simulaciones se ejecutan a través de seis scripts. El orden de ejecución dependerá de los modos que se quieran simular. Cada uno de dichos scripts corresponden a un modo de simulación de los descritos previamente, modo independiente, fusión y modo simultáneo, y a los PDOP correspondientes a cada uno de esos modos.

Observando la ilustración 79, se tienen los scripts 'lps_unico.m' y 'lps_pdop_unico.m' que hacen alusión al modo independiente; 'lps_fusion.m' y 'lps_pdop_fusion.m' para el modo en el que se fusionan los datos; y 'lps_triple.m' y 'lps_pdop_triple.m' que se refieren al modo simultáneo. En los scripts que no incluyen "pdop" en el nombre se realizan los cálculos de posicionamiento para ambas trilateraciones y en los scripts cuyo nombre incluye "pdop" se desarrollan los diferentes cálculos de PDOP de cada uno de los mismos modos.

Los modos independiente y triple se pueden ejecutar independientemente, lanzando los scripts 'lps_unico.m' y 'lps_triple.m' respectivamente, para calcular el posicionamiento en cada uno de dichos modos. Después, para realizar los respectivos cálculos de PDOP se ejecutarán 'lps_pdop_unico.m' y 'lps_pdop_triple.m'. Los datos de posicionamiento se guardan en el espacio de trabajo, cada modo guarda su propia matriz, por lo que se pueden ejecutar distintos modos sin perjuicio ninguno. En el caso del modo de fusión, hay una dependencia: se necesita ejecutar al menos una vez el script de posicionamiento del modo independiente, 'lps_unico.m', antes de ejecutar el script de posicionamiento del modo de fusión 'lps_fusion.m'. Después, al igual que en el resto de los modos, se ejecuta el script 'lps_pdp_fusion.m' para calcular el PDOP correspondiente.

Cada uno de los scripts de posicionamiento 'lps_unico.m' y 'lps_triple.m' tienen unos valores definidos de número y posición de balizas, límites del entorno, incrementos de posición y ruido del entorno en distintas variables; haciendo estos parámetros fácilmente configurables por el usuario al inicio del script. Los valores del script 'lps_fusion.m' vienen determinados por los valores del script 'lps_unico.m'.

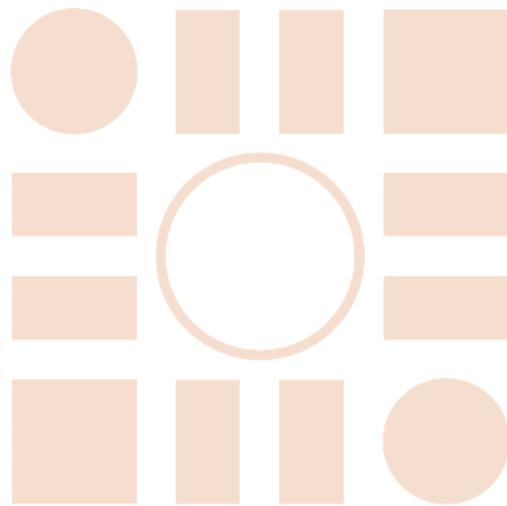
XI. BIBLIOGRAFÍA

- [1] Ureña Ureña, Jesús; Hernández Alonso, Álvaro; Gualda Gómez, David; Villadangos Carrizo, José Manuel; Mannay, Khaoula. "Adaptación de un sistema de posicionamiento ultrasónico para operación con drones a baja velocidad en zonas acotadas". *Actas de las XXXIX Jornadas de Automática (JA 2018)* (ISBN: 978-84-09-04460-3). 2018, p. 394-400.
- [2] Sabatini, R.; Moore, T.; Ramasamy, S. "Global navigation satellite systems performance analysis and augmentation strategies in aviation". *Prog. Aerosp. Sci.* 2017, 95, 45–98.
- [3] Q. Sultana, M. A. Hussain and M. A. Malik, "Investigations into the relationship between GPSDOP and unit-vector tetrahedron volume", *2013 International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT)*, Tiruvannamalai, 2013, pp. 1-4.
- [4] R. Rodríguez, "Utilización de los determinantes de Cayley-Menger en la localización por Trilateración esférica", *Trabajo de Investigación, Universidad de Alcalá, 2005-2006*, <http://platea.pntic.mec.es/~rrodrigu/>.
- [5] Frank Dellaert, "Sensor Fusion as Weighted Averaging", *Technical Report, Center for Robotics and Intelligent Machines, Georgia Institute of Technology*.
- [6] Mannay, Khaoula; Ureña Ureña, Jesús; Hernández Alonso, Álvaro; Gualda Gómez, David; Villadangos Carrizo, José Manuel. "Analysis of performance of Ultrasonic Local Positioning Systems for 3D Spaces". *Proc. of 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN 2017)*. (ISBN: 978-4-86049-074-4). 2017, p. 1-4.
- [7] Khaoula Mannay, Jesus Ureña, Álvaro Hernández, Mohsen Machhout. "Performance of Location and Positioning Systems: a 3D- Ultrasonic System Case". *Advances in Science, Technology and Engineering Systems Journal (ASTESJ)*. Volume 3, Issue 2, Page No 106-118, 2018.
- [8] D. Ruiz, E. Garcia, J. Ureña, D. de Diego, D. Gualda, and J. C. Garcia, "Extensive Ultrasonic Local Positioning System for navigating with mobile robots", *2013 10th Workshop on Positioning, Navigation and Communication (WPNC)*, 2013.
- [9] Ureña Ureña, Jesús; Villadangos Carrizo, José Manuel; Gualda Gómez, David; Pérez Rubio, María del Carmen; Hernández Alonso, Álvaro; García Domínguez, Juan Jesús; Jiménez Martín, Ana; García García, Juan Carlos; Arango Vargas, Juan Felipe; Díaz Llerena, Edel. "Technical Description of Locate-US: an Ultrasonic Local Positioning System based on Encoded Beacons.". *Proc. of 2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN 2016)*. (ISBN: 978-1-5090-2424-7). 2016, p. 1-4.
- [10] Phillips AH (1984) Geometrical determination of PDOP. *Navigation* 31:329–337.

[11] Ureña Ureña, Jesús; Lázaro Galilea, José Luis. “Sistemas de posicionamiento en interiores: tecnologías, aplicaciones y tendencias”. *Presentación del Departamento de Electrónica, Universidad de Alcalá, 2019.*

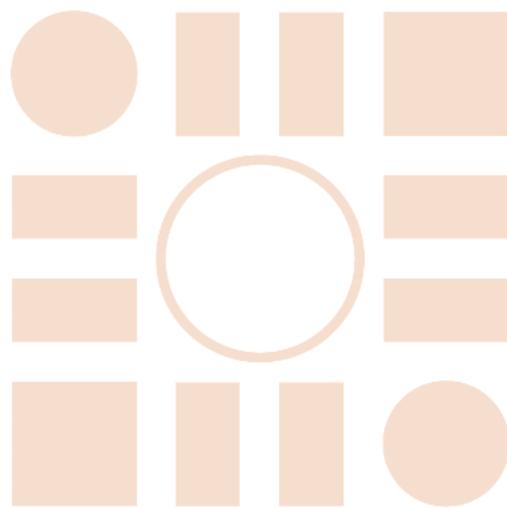
[12] García Domínguez, Juan Jesús; Mazo Quintas, Manuel; Ureña Ureña, Jesús; Ruiz Pereda, Francisco Daniel. “Location and Local Positioning Systems”. *Course of Department of Electronics; Master in Advanced Electronics Systems and Intelligent Systems; University of Alcalá.*

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá