



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE
TELECOMUNICACIONES
DEPARTAMENTO DE
ELECTROMAGNETISMO Y RADIACIÓN



**DETECCIÓN DE LA DIRECCIÓN DE ARRIBO USANDO TÉCNICAS
DE REDES NEURONALES**

DAVID SANTAMARIA
ROGHERT RIVERO

Bárbula, 6 de Julio del 2015



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE
TELECOMUNICACIONES
DEPARTAMENTO DE
ELECTROMAGNETISMO Y RADIACIÓN



**DETECCIÓN DE LA DIRECCIÓN DE ARRIBO USANDO TÉCNICAS
DE REDES NEURONALES**

TRABAJO ESPECIAL DE GRADO PRESENTADO ANTE LA ILUSTRE UNIVERSIDAD DE
CARABOBO PARA OPTAR AL TÍTULO DE INGENIERO DE TELECOMUNICACIONES

DAVID SANTAMARIA
ROGHERT RIVERO

Bárbula, 6 de Julio del 2015



UNIVERSIDAD DE CARABOBO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE
TELECOMUNICACIONES
DEPARTAMENTO DE
ELECTROMAGNETISMO Y RADIACIÓN



AVAL DEL TUTOR

Quien suscribe Ing. Rafael Alborno, titular de la cédula de identidad 3.835.443, en mi carácter de TUTOR del Trabajo Especial de Grado titulado:

Detección de la dirección de arribo usando técnicas de redes neuronales

Y presentado por los bachilleres DAVID SANTAMARIA, cédula de identidad 24.300.371, ROGHERT RIVERO, cédula de identidad 23.572.691, para optar al Título de Ingeniero de Telecomunicaciones, hago constar que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se le designe

Firma

Prof. ING. RAFAEL ALBORNOZ

CI: 3.835.443

TUTOR

Bárbula, 6 de Julio del 2015

Dedicatoria

Línea 1 de dedicatoria

Línea 2 de dedicatoria

Última línea de dedicatoria

DAVID SANTAMARIA

Línea 1 de dedicatoria

Línea 2 de dedicatoria

Última línea de dedicatoria

ROGHERT RIVERO

Agradecimientos

Escriba aquí los agradecimientos y las personas que hicieron posible el desarrollo y la culminación de su Trabajo Especial de Grado...

Índice general

Índice de Figuras	XI
Índice de Tablas	XV
Acrónimos	XVII
Resumen	XIX
I. El problema	1
1.1. PLANTEAMIENTO DEL PROBLEMA	1
1.2. JUSTIFICACION DE LA INVESTIGACION	4
1.3. OBJETIVOS	4
1.3.1. OBJETIVO GENERAL	4
1.3.2. OBJETIVOS ESPECIFICOS	4
1.4. ALCANCES	5
II. Marco Conceptual	7
2.1. Redes Móviles	7
2.1.1. GSM (Global System for Mobile Communication)	8
2.1.2. Acceso a la Red Móvil	8
2.1.2.1. SDMA	9
2.2. Antenas Inteligentes	11
2.2.1. Tipos De Antenas Inteligentes	12
2.2.1.1. Haz Conmutado	12
2.2.1.2. Haz de Seguimiento]	13
2.2.1.3. Haz Adaptativo	14
2.2.2. Arquitectura de un Sistema De Antena Inteligente.	15
2.3. Estimación de la Dirección de Arribo	18
2.3.1. Detección Del Ángulo De Llegada	18
2.3.2. Algoritmo Deterministic Maximum Likelihood (DML)	22
2.3.3. Algoritmo MUSIC DoA	23
2.4. Redes Neuronales	25
2.4.1. Clasificación De Redes Neuronales	28

2.4.2.	Tipos De Redes Neuronales	30
2.4.3.	Tipos De Función De Activación	32
2.4.4.	Redes Perceptron Multicapa	33
2.4.4.1.	Arquitectura del perceptrón multicapa	33
2.4.5.	Redes neuronales de base radial	35
2.4.5.1.	Arquitectura de Redes neuronales de base radial	35
III.	Procedimientos de la investigación	39
3.1.	Metodo a Desarrollar	39
3.1.1.	Creación de las Redes Neuronales	42
3.2.	Caracterización de los Vectores de Entrenamiento	45
3.3.	Programas de Entrenamiento y Evaluación	49
3.3.1.	Performance de las Redes Generadas	51
3.3.2.	Algoritmos MUSIC y DML	52
IV.	Análisis, interpretación y presentación de los resultados	55
4.1.	Estructura para comprensión de resultados	55
4.2.	Resultados en condiciones óptimas	57
4.2.1.	1 señal Incidente	57
4.2.2.	2 señales Incidentes separadas	61
4.2.3.	2 señales Incidentes con separación intermedia	64
4.2.4.	2 señales Incidentes próximas	67
4.2.5.	4 señales Incidentes	70
4.2.6.	7 señales Incidentes	73
4.3.	Resultados con relación señal a ruido de 20 dB	78
4.3.1.	1 señal Incidente	78
4.3.2.	2 señales Incidentes separadas	83
4.3.3.	2 señales Incidentes con separacion intermedia	85
4.3.4.	2 señales Incidentes próximas	88
4.3.5.	4 señales Incidentes	91
4.3.6.	7 señales Incidentes	94
4.4.	Resultados con relacion señal a ruido de 5 dB	99
4.4.1.	1 señal Incidente	99
4.4.2.	2 señales Incidentes separadas	103
4.4.3.	2 señales Incidentes con separación intermedia	106
4.4.4.	2 señales Incidentes próximas	109
4.4.5.	4 señales Incidentes	112
4.4.6.	7 señales Incidentes	115
V.	Conclusiones y recomendaciones	121
5.1.	Conclusiones	121
5.2.	Recomendaciones	123

A. Apendice A	125
1.1. Entrenamiento	125
1.1.1. Evaluación	128
1.2. Funcionamiento	131
B. Apendice B	135
2.1. Algoritmo MUSIC	135
2.2. Algoritmo DML	137
 Referencias Bibliográficas	 141

Índice de figuras

2.1. Acceso Múltiple por División de Espacio.	10
2.2. Comparación de Radiación.	11
2.3. Radiación de Haz Conmutado.	12
2.4. Radiación de Haz de Seguimiento.	13
2.5. Radiación de Haz Adaptativo.	14
2.6. Arquitectura de Arreglo de Antenas.	15
2.7. Ángulo de llegada para una apertura lineal.	19
2.8. Arreglo de M-elementos con D señales de arribo.	20
2.9. Red Neuronal Simple.	26
2.10. Red Hopfield.	28
2.11. Red Feedforward.	30
2.12. Red Hopfield.	31
2.13. Red Feedlateral.	31
2.14. Esquema de la red neuronal para DOA.	33
3.1. Esquemmatización.	40
3.2. Composición de la Red Neuronal MLP.	40
3.3. Composición de la Red Neuronal RBF.	41
3.4. Sistema de Detección cubriendo 360°.	42
3.5. Red MLP Backpropagation.	44
3.6. Arreglo de 10 dipolos con distancia de media longitud de onda entre ellos.	46
3.7. Diagrama de los pasos usados para entrenar cada red	49
3.8. Diagrama de los pasos usados para evaluar cada red	50
3.9. Diagrama del programa usado detectar la dirección de arribo	50
3.10. Detección de una señal con un ángulo de arribo de 15°	51
3.11. Detección de una señal con un ángulo de arribo de 6°	52
3.12. Detección de 2 señales con un angulo de arribo de -20° y 37°	52
3.13. Detección de 2 señales con un ángulo de arribo de -35° y 20°	53
4.1. MLP $\theta = -15$	57
4.2. MLP $\theta = 46$	58
4.3. RBF $\theta = -15$	58

4.4. RBF $\theta = 46$.	59
4.5. MUSIC $\theta = -15$.	59
4.6. MUSIC $\theta = 46$.	60
4.7. DML $\theta = -15$.	60
4.8. DML $\theta = 46$.	61
4.9. MLP $\theta_1 = -37$ y $\theta_2 = 48$.	62
4.10. RBF $\theta_1 = -37$ y $\theta_2 = 48$.	63
4.11. MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.	63
4.12. DML $\theta_1 = -37$ y $\theta_2 = 48$.	64
4.13. MLP $\theta_1 = -7$ y $\theta_2 = 8$.	65
4.14. RBF $\theta_1 = -7$ y $\theta_2 = 8$.	66
4.15. MUSIC $\theta_1 = -7$ y $\theta_2 = 8$.	66
4.16. DML $\theta_1 = -7$ y $\theta_2 = 8$.	67
4.17. MLP $\theta_1 = 24$ y $\theta_2 = 25$.	68
4.18. RBF $\theta_1 = 24$ y $\theta_2 = 25$.	69
4.19. MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.	69
4.20. DML $\theta_1 = 24$ y $\theta_2 = 25$.	70
4.21. MLP $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	71
4.22. RBF $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	72
4.23. MUSIC $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	72
4.24. DML $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	73
4.25. MLP $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	74
4.26. RBF $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	75
4.27. MUSIC $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	76
4.28. DML $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	77
4.29. MLP $\theta = -15$.	78
4.30. MLP $\theta = 46$.	79
4.31. RBF $\theta = -15$.	79
4.32. RBF $\theta = 46$.	80
4.33. MUSIC $\theta = -15$.	81
4.34. MUSIC $\theta = 46$.	81
4.35. DML $\theta = -15$.	82
4.36. DML $\theta = 46$.	82
4.37. MLP $\theta_1 = -37$ y $\theta_2 = 48$.	83
4.38. RBF $\theta_1 = -37$ y $\theta_2 = 48$.	84
4.39. MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.	84
4.40. DML $\theta_1 = -37$ y $\theta_2 = 48$.	85
4.41. MLP $\theta_1 = -7$ y $\theta_2 = 8$.	86

4.42. RBF $\theta_1 = -7$ y $\theta_2 = 8$.	86
4.43. RBF $\theta_1 = -7$ y $\theta_2 = 8$.	87
4.44. DML $\theta_1 = -7$ y $\theta_2 = 8$.	88
4.45. MLP $\theta_1 = 24$ y $\theta_2 = 25$.	89
4.46. RBF $\theta_1 = 24$ y $\theta_2 = 25$.	89
4.47. MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.	90
4.48. DML $\theta_1 = 24$ y $\theta_2 = 25$.	91
4.49. MLP $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	92
4.50. RBF $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	92
4.51. MUSIC $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	93
4.52. DML $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	94
4.53. MLP $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	95
4.54. RBF $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	96
4.55. MUSIC $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	97
4.56. DML $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$.	98
4.57. MLP $\theta = -15$.	99
4.58. MLP $\theta = 46$.	100
4.59. RBF $\theta = -15$.	100
4.60. RBF $\theta = 46$.	101
4.61. MUSIC $\theta = -15$.	101
4.62. MUSIC $\theta = 46$.	102
4.63. DML $\theta = -15$.	102
4.64. DML $\theta = 46$.	103
4.65. MLP $\theta_1 = -37$ y $\theta_2 = 48$.	104
4.66. RBF $\theta_1 = -37$ y $\theta_2 = 48$.	104
4.67. MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.	105
4.68. DML $\theta_1 = -37$ y $\theta_2 = 48$.	106
4.69. MLP $\theta_1 = -7$ y $\theta_2 = 8$.	107
4.70. RBF $\theta_1 = -7$ y $\theta_2 = 8$.	107
4.71. RBF $\theta_1 = -7$ y $\theta_2 = 8$.	108
4.72. DML $\theta_1 = -7$ y $\theta_2 = 8$.	109
4.73. MLP $\theta_1 = 24$ y $\theta_2 = 25$.	110
4.74. RBF $\theta_1 = 24$ y $\theta_2 = 25$.	110
4.75. MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.	111
4.76. DML $\theta_1 = 24$ y $\theta_2 = 25$.	112
4.77. MLP $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	113
4.78. RBF $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	113
4.79. MUSIC $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$.	114

4.80. DML $\theta_1 = -58, \theta_2 = -24, \theta_3 = 17$ y $\theta_4 = 58$	115
4.81. MLP $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$	116
4.82. RBF $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$	117
4.83. MUSIC $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$	118
4.84. DML $\theta_1 = -57, \theta_2 = -53, \theta_3 = -32, \theta_4 = -16, \theta_5 = 5, \theta_6 = 22$ y $\theta_7 = 53$	119

Indice de tablas

4.1. Variables	56
--------------------------	----

Acrónimos

DML	D eterministic M aximun L ikelihood
DOA	D ireccionl O f A rrival
DSP	D igital S ignal P rocessing
ML	M aximun L ikelihood
MUSIC	C lasificación de S eñal M ultiple
LED	L ight- E mitting D iode
OPSU	O ficina de P lanificación del S ector U niversitario
RN	R ed N euronal
SINR	S ignal to N oise plus I nterference R atio
SML	S tochastic M aximun L ikelihood
UC	U niversidad de C arabobo

DETECCIÓN DE LA DIRECCIÓN DE ARRIBO USANDO TÉCNICAS DE REDES NEURONALES

por

DAVID SANTAMARIA y ROGHERT RIVERO

Presentado en el Departamento de Electromagnetismo y Radiación
de la Escuela de Ingeniería en Telecomunicaciones
el 6 de Julio del 2015 para optar al Título de
Ingeniero de Telecomunicaciones

RESUMEN

El siglo XXI ha sido descrito como el inicio de la era de la información. La evolución de las telecomunicaciones, desde el teléfono alámbrico hasta los servicios de comunicación personal, es resultado del avance y de la disponibilidad de los servicios inalámbricos. Esta industria ha experimentado un explosivo crecimiento, en los diferentes servicios de telecomunicaciones, que a su vez son proporcionados en diferentes formas. Los servicios de comunicación inalámbrica son proporcionados por radio enlaces, donde la información tal como voz y datos son transmitidos por ondas electromagnéticas moduladas. Por lo tanto, todos los servicios de telecomunicaciones están sujetos a la interferencia de propagación del medio ambiente y del efecto de interferencia de las señales de los canales en un sistema de re-uso de

frecuencia. Las antenas inteligentes han surgido con la necesidad de brindar mayor y mejor servicio a los abonados, las cuales modificando su haz de radiación evitan interferencias y evitan errores introducidos por la presencia de ruido. En la última década se ha venido trabajando con una serie de algoritmos los cuales mediante la detección de la dirección de arribo de la señal (DOA), se han encargado de mejorar el rendimiento del sistema. Las redes neuronales son un sistema relativamente de nuevo desarrollo el cual, se muestra como una vía eficiente y conveniente para realizar la ardua tarea de dirigir y controlar este tipo de antena, debido a que estas redes están creadas precisamente para la toma de decisiones en ambientes cambiantes. ...

Palabras Claves: DOA, Redes, Neuronales

Tutor: ING. RAFAEL ALBORNOZ

Profesor del Departamento de Electromagnetismo y Radiación

Escuela de Telecomunicaciones. Facultad de Ingeniería

Capítulo I

El problema

1.1. PLANTEAMIENTO DEL PROBLEMA

La capacidad de cualquier sistema de comunicaciones puede ser incrementada directamente aumentando el ancho de banda existente de los canales de comunicación o permitiendo el uso de nuevas frecuencias para el servicio en cuestión. Sin embargo, el problema es que el espectro de frecuencias es finito y gran parte ya está utilizado por estaciones de radio y televisión, los militares, servicios de emergencia y transmisores de microondas; además, el medio electromagnético está congestionándose con la proliferación de fuentes de interferencia intencionales y no intencionales.

En lo que se refiere a las condiciones de propagación la señal sufre pérdidas de energía y cambios de fase por el medio en el cual se propaga, el cual es el espacio libre, la señal se propaga por reflexión, difracción y dispersión. Por otra parte, el efecto más adverso de la propagación al cual se enfrentan los sistemas de comunicaciones inalámbricas es el de desvanecimiento por multipropagación. El efecto de desvanecimiento por multipropagación, es causado por la superposición de señales reflejadas provenientes de varias señales del medio ambiente de propagación, creando errores en la transmisión digital. El uso del sistema de antenas inteligentes es con la finalidad de utilizar una de las últimas fronteras tecnológicas que promete

mejorar sustancialmente la capacidad de los sistemas inalámbricos. El sistema de antenas inteligentes ofrece la habilidad de mejorar la recepción, detección y parámetros de estimación de la señal. De estos quizá el más importante para la industria de las telecomunicaciones es el mejoramiento de la recepción de la señal, el cual está implícitamente relacionado con la dirección de arribo de la señal. La mejora en la capacidad se debe al incremento enorme de la ganancia de la antena hacia las señales deseadas y simultáneamente la reducción de la ganancia de la antena hacia los interferentes.

La estimación de la dirección de arribo (DOA) de las señales incidentes en un sistema de antenas inteligentes es crítica, porque es uno de los problemas principales en sistemas inalámbricos. Para estimar la DOA típicamente se realiza por medio de algoritmos de alta resolución (clasificación múltiple de señales, máxima verosimilitud, etc.) los cuales, en términos estadísticos, hacen uso únicamente de la información del primer y segundo momento (media y varianza) de los datos.

Los algoritmos que se emplean en la actualidad por sus altas prestaciones son, el algoritmo MUSIC (clasificación múltiple de señal), el cual se basa en el método de subespacios los cuales fundamentan en observaciones referentes a la autodescomposición de la matriz de correlación en un subespacio de señal y uno de ruido, como se explicara en secciones posteriores [1], y el algoritmo LM (maxima verosimilitud) el cual se presenta en la literatura de dos formas, en forma estocástica (SML, Stochastic ML) y en forma determinista (DML, Deterministic ML) [2].

Los algoritmos mencionados anteriormente son empleados por el sistema de antenas inteligentes para estimar la DOA. En este sistema se debe asumir que los elementos del arreglo sean idénticos y tengan un comportamiento lineal para que el desempeño de los algoritmos sea eficiente. En la mayoría de los casos, el comportamiento de este arreglo de sensores no es lineal o es desconocido, lo que implica que el comportamiento de los algoritmos empleados en el sistema de antenas se vuelva complejos y que su desempeño sea deficiente.

La importancia en todo ámbito de las redes neuronales, esta demarcada por su sencillez y por la facilidad de esta de adaptarse a diversas circunstancias.

Con las redes neuronales se intenta expresar la solución de problemas complejos, no como una secuencia de pasos, sino como la evolución de unos sistemas de computación inspirados en el funcionamiento del cerebro humano, y dotados por tanto de cierta “inteligencia” los cuales no son sino la combinación de gran cantidad de elementos simples de proceso (neuronas) interconectados que, operando de forma masivamente paralela, consiguen resolver problemas relacionados con el reconocimiento de formas o patrones, predicción, codificación, clasificación, control y optimización [3].

El tipo de inteligencia que plantea este tipo de antenas inteligentes se puede definir como de haz adaptativo debido a que esta constituye el máximo nivel de inteligencia que se podría dar a un sistema de antenas. En este sistema, las salidas de cada elemento del arreglo de antenas se ponderan con un factor de peso cuyo valor se asigna dinámicamente para conformar un diagrama de radiación que presente el haz principal hacia la posición del usuario deseado y los haces o lóbulos secundarios hacia las direcciones de las componentes de multitrayecto de la señal deseada y mínimos o nulos de radiación en las direcciones de las fuentes de interferencia [4].

1.2. JUSTIFICACION DE LA INVESTIGACION

Para mejorar la eficiencia de los algoritmos y con ello mejorar el desempeño del sistema de antenas inteligentes; se proponen las redes neuronales para realizar la función de los algoritmos tradicionales, además la idea principal es utilizar la característica de no linealidad de las redes neuronales con la finalidad desarrollar un modelo que desempeñe las funciones del sistema de antenas inteligentes eficientemente, sin asumir condiciones ideales de propagación.

Supone una investigación novedosa, dado que se busca optimizar el desempeño de las redes neuronales, determinando en que delimitadas circunstancias las redes neuronales se comportan de manera óptima y eficiente, aparte de diferenciar que tipos de redes neuronales logran este comportamiento.

La necesidad de detectar la dirección de arribo (DOA) de manera eficaz y eficiente viene de la mano con la creciente demanda del servicio inalámbrico, se pretende deducir si los sistemas de redes neuronales ofrecen una alternativa fiable a esta necesidad, aparte de analizar su desempeño con algoritmos ya mundialmente utilizados.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Desarrollar un modelo matemático que estime la dirección de arribo de las señales incidentes en un sistema de antenas inteligentes empleando una red neuronal con el fin de proporcionar un mejor desempeño del sistema de antenas inteligentes.

1.3.2. OBJETIVOS ESPECIFICOS

- Desarrollar distintos modelos de redes neuronales las cuales sean capaces de identificar y detectar las señales dirigidas a un arreglo de antenas.

- Evaluar las redes neuronales desarrolladas considerando condiciones no óptimas de entorno.
- Comparación de las redes neuronales desarrolladas con algoritmos de detección de arribo basados en métodos de sub espacios como MUSIC, al igual que otros basados en procesos estadísticos como LM.
- Estudio de la respuesta de los distintos tipos de redes neuronales a condiciones de entorno no favorables, tomando como referencia el desempeño de los algoritmos MUSIC y LM.
- Exponer los resultados de qué tipo de red neuronal, es óptima en determinada circunstancia no favorables de entorno.

1.4. ALCANCES

En el proceso de desarrollo de los diferentes aspectos de este trabajo, se pretende lograr la habilidad necesaria en el manejo de los programas los cuales se utilizaran para llevar a buen término los objetivos del mismo.

Una de las plataformas donde se espera desenvolver el análisis de las redes neuronales es la herramienta de software matemático "*MATLAB*" el cual cuenta con un toolbox de redes neuronales (Neural Network Toolbox), este toolbox al encontrarse totalmente orientado a redes neuronales permite usarlo en aplicaciones orientadas a datos de adaptación, reconocimiento de patrones, la agrupación, la predicción de series temporales, y modelado de sistemas dinámicos y control.[5]

Empleando esta herramienta de cómputo se pretende desarrollar y optimizar una red neuronal la cual sea capaz de detectar la dirección de arribo de una señal o conjunto de señales. Se generaran simulaciones sucesivas de aprendizaje y se evaluaran los resultados hasta obtener una red que cumpla con las especificaciones planteadas.

Siguiendo este mismo orden de ideas para lograr una comparación aceptable de la red neuronal desarrollada, se manejaran las herramientas que posee MATLAB

para generar los algoritmos de obtención de DOA más comunes y frecuentemente utilizados, entre ellos MUSIC, y el algoritmo LM.[\[6\]](#)

Se procurará que los algoritmos se lleven a cabo en situaciones que asemejen en todo aspecto a las condiciones normales de uso, con señales interferentes, ruido aditivo, y más de una fuente de origen para evaluar su respuesta en un entorno real y cotidiano.

Capítulo II

Marco Conceptual

2.1. Redes Móviles

Todos los sistemas de comunicación tienen fundamentalmente el mismo objetivo: transmitir la máxima cantidad de información con el mínimo número de errores. Los modernos sistemas de comunicaciones inalámbricas digitales no son una excepción. Dada cualquier entrada digital, el codificador de fuente elimina la redundancia en los bits de información, lo que maximiza la cantidad de la información útil transferidos en el sistema de comunicación. La salida del generador de código fuente es procesada por el codificador de canal, que incorpora información de control de errores en los datos para reducir al mínimo la probabilidad de error en la transmisión. La salida del codificador de canal es procesada más adelante por la unidad de procesamiento de señales digitales, a fin de permitir la comunicación simultánea de muchos usuarios. Un ejemplo de esto sería la formación de haz digital, que mediante el uso de las propiedades geométricas del arreglo de antenas, es capaz de concentrar las señales de múltiples usuarios en diferentes direcciones deseadas, lo que permite a más usuarios ser atendidos por el sistema. El flujo de datos generado es luego procesado por el modulador que se encarga de desplazar la señal en banda base a su entrada en la versión pasa-banda en la salida, debido a las limitaciones de ancho de banda del sistema de comunicación. La secuencia de la

información generada en la salida del modulador luego se introduce en el arreglo de antenas y se transmiten a través del canal inalámbrico.

2.1.1. GSM (Global System for Mobile Communication)

El sistema GSM (Global System for Mobile Communication) puede ser definido como un sistema de radio comunicaciones, porque está destinado para comunicaciones de media y larga distancia, digital, móvil y celular, lo interesante del sistema es que nos proporciona servicios de datos, e incluye integración de servicios. Este sistema nos proporciona una calidad de audio que ayuda a la fiabilidad de la comunicación, incluye sistemas de seguridad en el aspecto de autenticación de usuarios actualmente este sistema es utilizado a nivel mundial por la mayoría de operadores celulares.

GSM utiliza una combinación de los esquemas de acceso múltiple FDMA y TDMA en su espectro de frecuencias, que tiene una anchura de 25 MHz. FDMA divide el ancho de banda de 25 MHz en 124 frecuencias portadoras de 200 KHz cada una. Cada canal de 200 KHz se divide entonces en ocho ranuras de tiempo empleando TDMA. Los sistemas GSM manejan velocidades de datos de 9.6 Kbps y la función SMS. Esto proporciona la posibilidad de hablar y escuchar al mismo tiempo, aunque para ello solo se emplea una antena. Y además, facilita enormemente la transmisión de datos, y todos los servicios que ese tipo de datos aportan, como son los mensajes de texto, fax, internet [7].

2.1.2. Acceso a la Red Móvil

Para lograr que múltiples usuarios accedan a la red, se utilizan tres técnicas principales de acceso, las cuales son:

- El estándar de acceso múltiple por división de frecuencia (FDMA)
- Acceso múltiple por división de tiempo (TDMA)

- Sistemas de acceso Múltiple por división de código (CDMA)

Sin duda son los métodos de acceso a la red que se han utilizado durante muchos años, e incluso algunos han mejorado para establecer su existencia, hoy en día se habla de un nuevo método de acceso que se encuentra en desarrollo y en algunos casos ya ha sido implantado, como lo es SDMA (Acceso Múltiple por División de Espacio).

2.1.2.1. SDMA

El método de Acceso Múltiple por División de Espacio es una de las últimas fronteras que están siendo implementadas en los nuevos sistemas de comunicaciones móviles. Ésta permite realizar una transmisión y recepción espacialmente selectiva. SDMA, se basa en la tecnología de Antenas Inteligentes empleadas en las estaciones bases de comunicaciones móviles [2.2](#).

En este sentido, como cada usuario móvil ocupa una localización espacial única, esta nueva técnica de acceso permite que múltiples usuarios que se encuentren angularmente espaciados, puedan acceder a la red, al mismo tiempo y con la misma frecuencia (en caso de FDMA) o con el mismo código (en caso de CDMA) sin causar interferencia unos con otros, de modo que se trata de un híbrido de las técnicas de acceso múltiple existentes y la nueva técnica SDMA. Esto es posible, puesto que al habilitar el diagrama de radiación para transmisión/recepción en un sector específico, existe la posibilidad de incluir en ese instante de tiempo a múltiples usuarios que estén ubicados en dicho sector, a través de los métodos TDMA, FDMA o CDMA.

Adicionalmente, todo el beneficio de SDMA se traduce en un aumento de la capacidad del sistema. La capacidad de un sistema de comunicaciones móviles se puede definir como la tasa binaria que puede ofrecerse en el ancho de banda disponible y en un área geográfica determinada, es decir bist/s/Hz/m^2 .

La realización de esta técnica de filtrado espacial se realiza mediante Antenas Inteligentes, que son sistemas capaces de modificar el tiempo, la frecuencia y la respuesta espacial [8].

Con SDMA, varios teléfonos móviles pueden compartir la misma frecuencia dentro de una celda. Múltiples señales que llegan a la estación base pueden ser separadas por el receptor de la estación base, siempre que su separación angular sea mayor que el ancho de haz de transmisión / recepción. Esto se muestra en la figura 2.1. Los haces que tienen la misma sombra usan la misma banda de frecuencias. Esta técnica es llamada reutilización del canal vía separación angular.

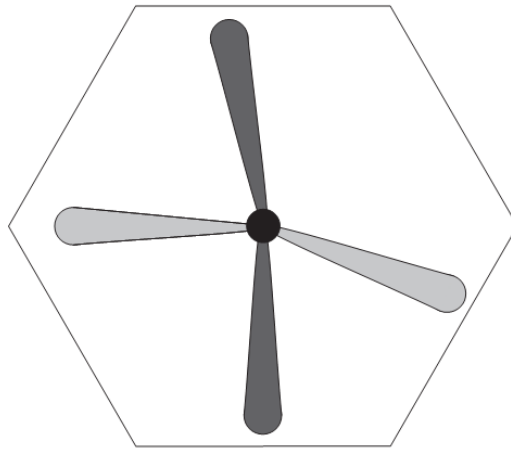


Figura 2.1: Acceso Múltiple por División de Espacio.

Para la alta tasa de datos y sistemas de comunicación móvil de alta confiabilidad, los métodos activos contra el desvanecimiento son requeridos. Un sistema SDMA es una medida eficaz para hacer frente al desvanecimiento, ya que, distingue a las señales de radio en el dominio espacial o angular utilizando la directividad de la antena o la formación de haz de acuerdo con la dirección de llegada (DOA) de las señales [6].

2.2. Antenas Inteligentes

Las antenas inteligentes son aquellas que poseen la capacidad de ajustar su diagrama de radiación de forma predefinida. El sistema funciona de tal forma que cuando el usuario se desplaza, se modifica la dirección del lóbulo para que se mueva con él.

La razón fundamental para usar antenas inteligentes en una red inalámbrica es que se puede incrementar su capacidad, esto es, la posibilidad de manejar más usuarios por estación base. Dicha capacidad depende, principalmente, del nivel de interferencia comparado con la potencia recibida (SINR), ya que la principal fuente de ruido en estos sistemas es la interferencia de otros usuarios.

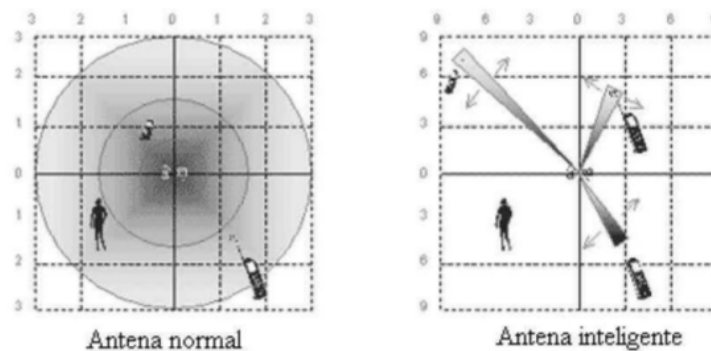


Figura 2.2: Comparación de Radiación.

Una antena inteligente es la combinación de un arreglo de antenas con una unidad de procesamiento digital de señales (DSP) que optimiza los diagramas de transmisión y recepción dinámicamente en respuesta a una señal de interés en el entorno. Es aquella que, en vez de disponer de un diagrama de radiación fijo, es capaz de generar o seleccionar haces muy directivos enfocados hacia el usuario deseado, e incluso adaptarse a las condiciones radioeléctricas en cada momento [9].

2.2.1. Tipos De Antenas Inteligentes

2.2.1.1. Haz Conmutado

Es la configuración más simple de Antenas Inteligentes. El sistema genera varios haces a ángulos prefijados que se van conmutando secuencialmente dando como resultando un barrido discreto de la zona de cobertura en posiciones angulares fijas. En cada posición discreta del haz se activa el sistema de recepción para detectar la posible existencia de señales. En caso de recibir señal, el sistema guarda información correspondiente a la posición del haz (ángulo + identificación de usuario) y se establece la comunicación con el usuario en un intervalo de tiempo. Después de este intervalo se conmuta al siguiente haz para detectar la existencia de otros posibles usuarios hasta llegar al límite angular de la zona de cobertura. Este proceso se repite permanentemente en el tiempo.

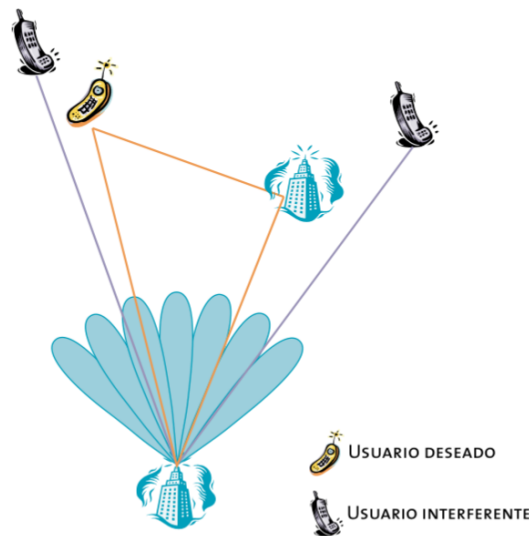


Figura 2.3: Radiación de Haz Conmutado.

Esta técnica no garantiza que el móvil se encuentre la dirección de máxima radiación del haz que le da servicio, ni que las señales interferentes se vean notablemente reducidas (ya que siempre es posible que alguna entre por uno de los lóbulos secundarios). De hecho, sería posible recibir una señal interferente por un punto

del diagrama de radiación con mayor ganancia que la señal deseada, empeorando apreciablemente las prestaciones del sistema.

2.2.1.2. Haz de Seguimiento]

Este sistema es un poco más complejo que el anterior. Está conformado por un arreglo de antenas con una red de excitación que permite controlar electrónicamente las fases de las corrientes de excitación que llegan a los elementos del arreglo para modificar la dirección del haz convenientemente y establecer comunicación con el usuario respectivo. A diferencia del sistema de haz conmutado, el sistema haz de seguimiento ejecuta algoritmos DoA (Direction of Arrival) para identificar la dirección de arribo de las señales de los usuarios. Otra diferencia es que los cambios de fase para en el sistema conmutado se realizan a ángulos fijos, es decir corresponden a ángulos prefijado en el sistema y en el sistema de Haz de seguimiento el posicionamiento del haz tiene mayor resolución angular.

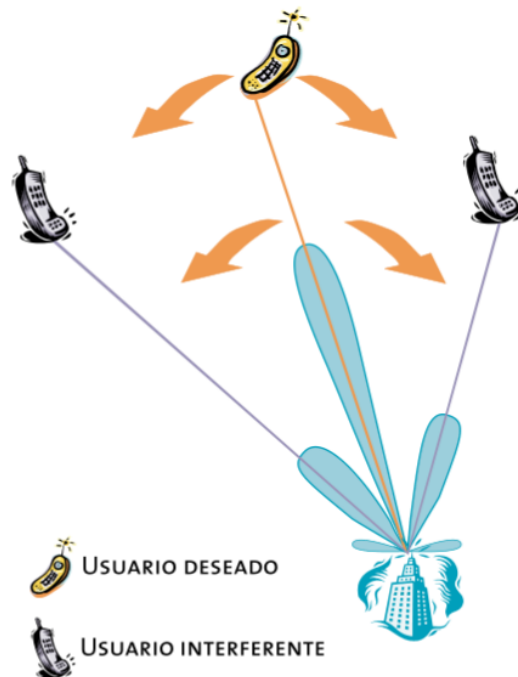


Figura 2.4: Radiación de Haz de Seguimiento.

Con esta técnica sí se puede garantizar que el usuario se encuentra iluminado en todo momento por el lóbulo principal y con máxima ganancia (dentro de las limitaciones de los algoritmos que se empleen). Sin embargo, tampoco puede evitarse que las interferencias entren por algún lóbulo secundario del diagrama de radiación. Para aprovechar las señales multitrayecto sería necesario detectar y seguir con otros haces de dichas componentes y luego procesarlas con un receptor Rake.

2.2.1.3. Haz Adaptativo

En este caso, la salida de cada elemento del arreglo se pondera con un factor de peso cuyo valor se asigna dinámicamente, de modo que se conforma el diagrama de radiación para maximizar algún parámetro de la señal (por ejemplo, la SINR). De este modo, el diagrama sintetizado habitualmente presentará un lóbulo principal en la dirección del usuario deseado, lóbulos secundarios en las direcciones de las componentes multitrayecto y mínimos (e incluso nulos) de radiación en las direcciones de las fuentes de interferencia. Esta técnica requiere el uso de complica-

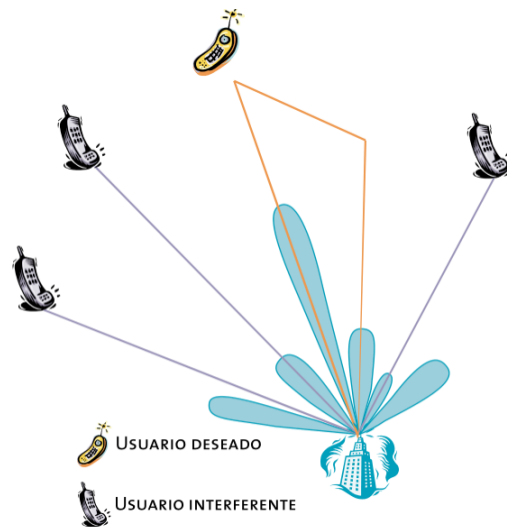


Figura 2.5: Radiación de Haz Adaptativo.

dos algoritmos, tanto para la detección de las señales deseada e interferente como para la optimización de los pesos que conforman el haz. Estos algoritmos suelen

conllevar una gran carga computacional, mientras que deben procesarse en tiempo real, por lo que suponen una seria limitación [10].

2.2.2. Arquitectura de un Sistema De Antena Inteligente.

El número de elementos en el arreglo debe ser relativamente bajo (el mínimo requerido), a fin de evitar la alta complejidad innecesaria en la unidad de procesamiento de señales. Los arreglos de antenas puede ser de uno, dos y tres dimensiones, dependiendo de la dimensión de espacio que se quiera acceder. La figura 2.6 muestra diferentes geometrías de arreglos que se pueden aplicar en las implementaciones de antenas adaptativas.

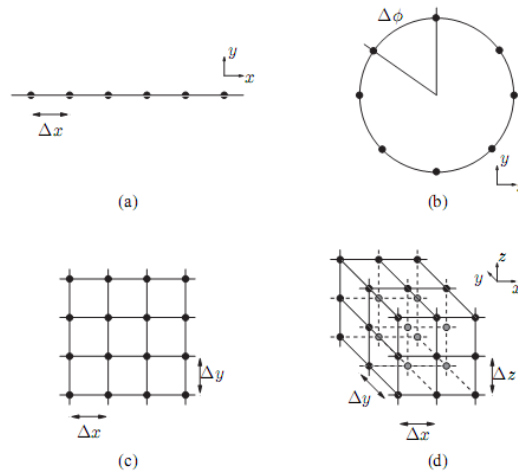


Figura 2.6: Arquitectura de Arreglo de Antenas.

La primera estructura se utiliza principalmente para la formación de haz en el único plano horizontal (azimut). Esto normalmente será suficiente para entornos al aire libre, al menos en las células grandes. El primer ejemplo (a) muestra un arreglo lineal unidimensional con el espaciamiento uniforme de los elementos de Δx . Esta estructura puede llevar a cabo la formación de haz en un plano dentro de un sector angular. Esta es la estructura más común debido a su baja complejidad. El segundo ejemplo (b) muestra una arreglo circular con espaciamiento angular uniforme entre elementos adyacentes de $\Delta \phi = 2\pi/M$, donde M representa el número de elementos. Esta estructura puede llevar a cabo la formación de haz en cualquier

dirección, pero, debido a su simetría, es más apropiado para la formación de haz azimutal. Las últimas dos estructuras se utilizan para realizar la formación de haz de dos dimensiones, en ambas azimutal y ángulos de elevación. Estas especificaciones son generalmente deseables para entornos urbanos interiores o densos. La vista frontal de un arreglo rectangular de dos dimensiones con un espaciamiento de los elementos horizontales de Δ_x y un espaciamiento de los elementos verticales de Δ_y se muestra en (c). La formación de haz en todo el espacio, dentro de todos los ángulos, requiere algún tipo de estructura cúbica o esférica (configuración tridimensional). El cuarto ejemplo (d) muestra una estructura cúbica con separaciones elemento de Δ_x , Δ_y , y Δ_z , respectivamente, en cada dirección en el espacio.

A pesar de que las antenas inteligentes son principalmente una tecnología de radio, estas inevitablemente pondrán nuevas demandas a las funciones de red; tales como los recursos y la gestión de movilidad. SDMA implica diferentes usuarios utilizando el mismo canal de comunicación física en la misma celda, separado sólo por ángulo. Cuando las colisiones angulares se producen entre estos usuarios, una de ellas con rapidez debe cambiar a otro canal para que la conexión no se rompa.

Para que la antena inteligente pueda obtener una ganancia razonable, es necesario un arreglo de antena con varios elementos. Normalmente los arreglos consisten de seis a diez elementos horizontales separados; los cuales, se han sugerido para entornos móviles al aire libre. La separación necesaria entre elementos es 0.4 a 0.5 longitudes de onda. Una antena de ocho elementos, por ejemplo, sería de aproximadamente 1,2 metros de ancho a una frecuencia de 900 MHz y 60 cm a 2 GHz. Con una creciente demanda del público para menos estaciones base visibles, las geometrías con el tamaño de varias longitudes de onda (que corresponden a las frecuencias portadoras que se utiliza actualmente), aunque no excesivo, podrían constituir un problema [6].

Las antenas inteligentes pueden tener sus sensores en diferentes distribuciones espaciales. En general, por motivos de simplicidad, las geometrías lineal y circular son los más utilizados. Un arreglo lineal de sensores, a pesar de ser los más simples a ser considerados, presenta el inconveniente de ambigüedad en sus estimaciones debido a la simetría de su diagrama de radiación, en relación al eje sobre el que se

colocan las antenas. En el caso de un arreglo circular, tal ambigüedad es suprimida, pero con la pérdida de la simplicidad del modelo.

El espaciamiento entre los elementos, es un parámetro de extrema importancia en el diseño de un conjunto de antenas. Para que se garantice una completa recuperación de las señales incidentes, evitando así una aparición de lóbulos indeseables, debemos asegurar que los elementos estén separados de una distancia mínima de $\lambda/2$, donde λ corresponde a la longitud de onda, valiendo aproximadamente $3 \times 10^8 / f_c$, donde f_c representa la frecuencia de la portadora, para los casos de ondas electromagnéticas que se propagan en el vacío. Esta consideración referente al espaciamiento mutuo entre los elementos es denominada la versión espacial del teorema del muestreo.

Una vez armado el arreglo de antenas, es importante resaltar que el campo eléctrico medido en el sensor m -ésimo es:

$$\mathbf{E} = S(t) \exp^{j[\omega - k(X_m \cos \theta + Y_m \sin \theta)]} \quad (2.1)$$

Siendo k el vector de onda

Para que todos los elementos del arreglo se comporten exactamente igual, se necesita que cada uno tenga una ganancia unidad en todo el ancho de banda de la señal y en consecuencia, la salida que resulta será proporcional al campo eléctrico recibido. Con esta condición, y descartando por conveniencia el término de la portadora pues en la práctica, generalmente, la señal es reducida a banda base antes del muestreo, entonces la señal de salida $X_m(t)$ del m -ésimo elemento se modela por:

$$X_m(t) = S(t) \exp^{-jk(X_m \cos \theta + Y_m \sin \theta)} \quad (2.2)$$

Para un arreglo de antenas de M elementos con geometría arbitraria, el vector de salida del arreglo se obtiene como:

$$\mathbf{x}(t) = \mathbf{a}(\boldsymbol{\theta})S(t) \quad (2.3)$$

Donde $\mathbf{a}(\theta)$ es llamado vector de direccionamiento, o vector de propagación del arreglo, el cual representa la respuesta del arreglo dada una dirección de llegada θ [11].

2.3. Estimación de la Dirección de Arribo

La Dirección de Arribo (DoA) de las señales recibidas por un arreglo de antenas adaptativas se calcula mediante un arreglo de antenas que trabaja con algoritmos inteligentes. Con el algoritmo de Estimación de Dirección de Arribo se determina la dirección de las señales recibidas que es la base para poder usar los arreglos de antenas inteligentes.

2.3.1. Detección Del Ángulo De Llegada

Para poder entender y realizar en forma clara la detección de los ángulos de llegada, es de vital importancia conocer y definir el escenario en el que se trabajará, el tipo de onda que se adapte a nuestras necesidades, y a partir de este formular un modelo matemático en el que podamos trabajar. Detectar el ángulo de llegada no es más que un problema de estimación de un parámetro que irá implícito en la información que recibamos por el arreglo en forma de desfases entre los diferentes sensores. A mayor número de elementos en la apertura, mayor resolución. Tenemos que tener en consideración que se trabajará con un arreglo lineal uniforme (ULA) consistente de M_s elementos, espaciados a una distancia d . Las antenas se unen omnidireccionales, y la ganancia de cada antena se normaliza a uno. La detección se basa en estimar el del ángulo de incidencia. Supongamos que tenemos un arreglo lineal equiespaciado o ULA. Si miramos el arreglo desde arriba, definimos θ_i como el ángulo de llegada para la señal i .

Para que se garantice una versión espacial del teorema de muestreo, y asegurar la recuperación unívoca de señales recibidas por el arreglo, y para fines de simulación serán considerados valores de espaciamiento d iguales a $\lambda/2$.

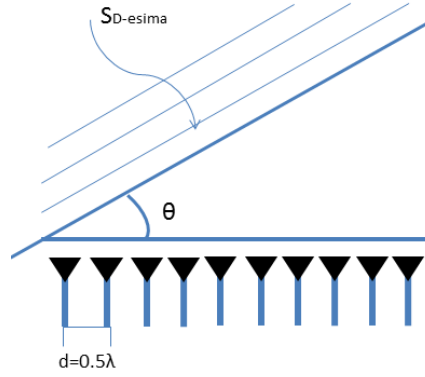


Figura 2.7: Ángulo de llegada para una apertura lineal.

El escenario en el que trabajaremos será con frentes de onda planos, ya que, estos son los únicos que emiten energía de interés y además son necesarios para que la única diferencia entre la señal de un sensor y otro sea un desfase de tiempo, que será la única información útil para calcular el ángulo de llegada

Así, para un ULA de M_s elementos, el vector de direccionamiento toma la siguiente forma:

$$\bar{a}(\theta_i) = \begin{bmatrix} 1 & \exp^{-jk d \cos \theta_i} & \exp^{-jk 2 d \cos \theta_i} & \dots & \exp^{-jk (M-1) d \cos \theta_i} \end{bmatrix}_M \quad (2.4)$$

Se asume que θ es un escalar real, perteneciente a $[0^\circ, 180^\circ]$. Esta es una restricción de los ULA's, pues las ondas planas proveniente de dos fuentes simétricas con respecto al eje del arreglo arriban al mismo en iguales instantes de tiempo, por lo que no se puede distinguir una de la otra [11] (ejemplo si llega una a 60° y otra a 300°). Esta ambigüedad de los ULA's se elimina mediante el uso de sensores que sólo procesan señales cuyas DoA's estén entre $[0^\circ, 180^\circ]$, para condicionar el proceso de detección, se trabajara con θ comprendidos entre $[0^\circ, 120^\circ]$, para un mejor manejo de sectorización.

Existen varios métodos para definir el pseudo-espectro a través de la formación de haz, la matriz de correlación, análisis Eigen, predicción lineal, MVDR (Minimum Variance Distortion-less Response), MLM (Maximum Likelihood Method), min- norm, MUSIC, root-MUSIC, CLOSET, ESPRIT, WSF, etc.

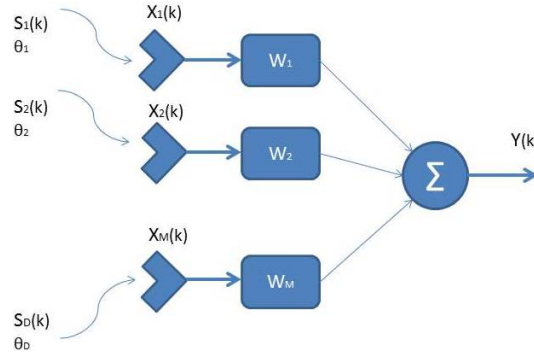


Figura 2.8: Arreglo de M-elementos con D señales de arribo.

En la Figura 2.8, D representa las señales con diferentes ángulos de llegada, que son recibidos por un arreglo de M antenas ponderadas. Cada señal recibida $x_m(k)$ contiene un nivel de ruido denominado ruido Gaussiano, ya que su densidad de probabilidad corresponde a una distribución de Gauss y por lo tanto es de promedio cero. Este ruido Gaussiano se introduce inevitablemente en el medio de transmisión ya que es inherente a este. La salida del arreglo está dado por $y(k)$ que puede ser escrita de la siguiente forma:

$$y(k) = W^T x(k) \quad (2.5)$$

$$x(k) = \begin{bmatrix} \bar{a}(\theta_1) & \bar{a}(\theta_2) & \dots & \bar{a}(\theta_D) \end{bmatrix}_{M \times D} \begin{bmatrix} s_1(k) \\ s_2(k) \\ \vdots \\ s_D(k) \end{bmatrix}_D + \begin{bmatrix} n_1(k) & n_2(k) & \dots & n_M(k) \end{bmatrix}_M \quad (2.6)$$

$$\bar{X} = A \cdot \bar{S} + \bar{N} \quad (2.7)$$

O bien podría escribirse en el tiempo discreto k como:

$$\bar{x}(k) = A \cdot \bar{s}(k) + \bar{n}(k) \quad (2.8)$$

Donde:

- $k=1,2,3,\dots,K$ (K número de muestras total en el tiempo).
- $W = [w_1 w_2 \dots w_M]^T$ vector pesos del arreglo.
- $\bar{x}$ = el vector señal recibida, de dimensión M
- $\bar{a}(\theta_i)$ = vector de arreglo para la dirección θ_i , de dimensión M
- A = es la matriz de arreglo para la dirección de arribo θ_i , de dimensión $M \times D$
- $\bar{S}$ = es el vector de señales incidentes monocromáticas en el tiempo k , de dimensión D
- $\bar{N}$ = el vector de ruido en cada antena con promedio cero y varianza σ^2 en el tiempo k , de dimensión M [12].

Tomando en cuenta la ecuación 2.8, si los transmisores están en movimiento la matriz de vectores de dirección varía con el tiempo y los correspondientes ángulos de arribo también cambian. A menos que se indique lo contrario, la dependencia del tiempo puede suprimirse y la matriz de correlación quedaría dada por [12]:

$$R_{xx} = \mathbf{E} [\bar{x} \bar{x}^H] \quad (2.9)$$

$$R_{xx} = \bar{A} R_{ss} \bar{A}^H + R_{nn} \quad (2.10)$$

Donde:

- R_{xx} es la matriz de autocorrelación de las señales recibidas por el arreglo de antenas $\bar{x}_m(k)$.
- R_{ss} es la matriz de autocorrelación de las señales incidentes $\bar{S}_D(k)$ en el arreglo de antenas.
- R_{nn} es la matriz de autocorrelación de ruido también definida como $\sigma_n^2 \bar{I}$.
- $\bar{I}$ Matriz identidad $M \times M$
- $\bar{Z}^H$ es el Hermitiano de la Matriz $\bar{Z}$
- D : es el número de señales que arriban al arreglo de antenas.
- M : es el número de elementos del arreglo.

2.3.2. Algoritmo Deterministic Maximum Likelihood (DML)

El modelo más conocido y frecuentemente usado basado en técnicas de estimación en procesamiento de señales es el de probabilidad máxima (Maximum Likelihood ML). El método ML asume un modelo para las señales y también una estructura estadística para la generación de información. Se han desarrollado varias versiones de Máxima Probabilidad (ML), entre ellas podemos mencionar la Máxima Probabilidad Determinística (Deterministic Maximum Likelihood DML), Máxima Probabilidad Estocástica (Stochastic Maximum Likelihood SML) y Máxima Probabilidad Cuadrática Iterativa (Iterative Quadratic Maximum Likelihood IQML); esta última necesita una estructura especial de arreglo. En esta sección se explicará el modelo DML, en el cual se asume como si la señal fuese determinística. Usando el modelo de la ecuación 2.19, la función logarítmica likelihood puede expresarse como [13] [14]:

$$L(\bar{x}(k), \sigma^2, \bar{s}(k), \theta) = -(M) \log(\sigma^2) - \frac{1}{\sigma^2} (\bar{x}(k) - \bar{s}(k))^2 \quad (2.11)$$

La estimación ML se obtiene maximizando $L(\bar{x}(k), \sigma^2, \bar{s}(k), \theta)$, fijando $\bar{x}$, $\bar{s}$ y θ , mientras maximizamos con respecto a σ^2 , así se llega a:

$$\sigma^2 = \frac{1}{M} |\bar{x}(k) - A \cdot \bar{s}(k)|^2 \quad (2.12)$$

Sustituyendo este valor en la ecuación 2.11 nos queda la función de estimación como:

$$\theta_{\text{DML}} = \underset{\theta}{\operatorname{argmin}} \operatorname{Tr} \left\{ P_A^\perp R_{xx} \right\} \quad (2.13)$$

Siendo $P_A^\perp = AA^\dagger$ la proyección ortogonal en el espacio nulo

Donde $A^\dagger = (A^H A)^{-1} A^H$ es la pseudo-inversa de A

$\operatorname{Tr} \{\}$ es la traza de una matriz [13].

2.3.3. Algoritmo MUSIC DoA

La clasificación de múltiple señal (MUSIC) es un popular método de estructura propia de alta resolución. MUSIC proporciona estimaciones objetivas del número de señales, el ángulo de llegada, y las fortalezas de las formas de onda. Desde su presentación el algoritmo MUSIC, en inglés Multiple Signal Classification, se ha considerado como una referencia de comparación para las demás técnicas de estimación de DOA.

El algoritmo MUSIC, se basa en la descomposición de la matriz de señales incidentes en un arreglo por medio del análisis del modelo de datos que aborda el mismo. En líneas generales, explora la geometría de los subespacios generados a partir de la matriz de correlación de las señales incidentes R_{xx} . Por este motivo, el método MUSIC es conocido como un enfoque geométrico para el problema de estimación de parámetros. Muchos de los algoritmos DOA confían en la matriz de correlación de arreglo. A fin de comprender la matriz de correlación de arreglo, vamos a empezar con una descripción del arreglo, la señal recibida, y el ruido aditivo [6].

Si tenemos D señales que llegan de D direcciones y son recibidas por arreglo de M elementos con M pesos de potencia como se observa en la figura 2.8. Cada una de las señales recibidas $x_m(k)$, adicional incluye ruido gaussiano y media cero.

MUSIC divide el espacio en 2 subespacios: La matriz de correlación (R_{xx}) está formada por una parte de ruido (R_{vv}) una matriz de dimensión $M \times (M - D)$ y la matriz del subespacio de la señal (R_{ss}) de dimensión $M \times D$.

Así tenemos D vectores propios asociados con las señales y $M - D$ vectores propios asociados con el ruido [9].

Los vectores propios del subespacio de ruido son ortogonales a la matriz de vectores de dirección en el ángulo de llegada. Los valores propios de R_{xx} son los valores $\lambda_0 \dots \lambda_{M-1}$, de tal manera que:

$$|R_{xx} - \lambda_i I| = 0 \quad (2.14)$$

Usando la ecuación R_{xx} podemos describir esto como:

$$|AR_{ss}A^H + \sigma_n^2 I - \lambda_i I| = |AR_{ss}A^H - (\lambda_i - \sigma_n^2) I| \quad (2.15)$$

Siendo entonces los valores propios n_i de $AR_{ss}A^H$:

$$n_i = \lambda_i - \sigma_n^2 \quad (2.16)$$

Se procede a encontrar los eigenvalores de ruido para R_{xx} . Para señales no correlacionadas los eigenvalores más pequeños son igual a la varianza del ruido. Se usan los valores más pequeños porque a este subespacio es el más reducido, lo que se puede traducir en mayor velocidad de procesamiento. Ahora se construye la matriz $M \times (M - D)$ con eigenvectores (vectores propios) como el siguiente [12]:

$$E_N = \begin{bmatrix} \bar{e}_1 & \bar{e}_2 & \bar{e}_{M-D} \end{bmatrix} \quad (2.17)$$

Desde la dirección de los vectores correspondientes; a los componentes de la señal, que son ortogonales a los vectores propios del subespacio de ruido es $\mathbf{a}(\boldsymbol{\theta})^H \mathbf{E}_N$ $\mathbf{a}(\boldsymbol{\theta}) \mathbf{E}_N^H = 0$, para $\boldsymbol{\theta}$ correspondiente a la DoA de un componente multitrayecto. Entonces las DOAs de las múltiples señales incidentes, puede ser estimado por la localización de los picos de un espectro espacial MUSIC dado por:

$$P_{\text{MUSIC}}(\boldsymbol{\theta}) = \left| \frac{1}{(\mathbf{a}(\boldsymbol{\theta})^H \mathbf{E}_N)(\mathbf{a}(\boldsymbol{\theta}) \mathbf{E}_N^H)} \right| \quad (2.18)$$

Siendo:

$\mathbf{E}_N$: La matriz del subespacio de ruido de dimensión $M \times (M - D)$.

$\mathbf{a}(\boldsymbol{\theta}_i)$: el vector de dirección en $\boldsymbol{\theta}$ direcciones de llegada, para $i=1,2,3,\dots,D$ [9].

2.4. Redes Neuronales

“Redes neuronales artificiales, son redes interconectadas masivamente en paralelo de elementos simples (usualmente adaptativas) y con organización jerárquica, las cuales intentan interactuar con objetos del mundo real del mismo modo que lo hace el sistema biológico”[15].

Una red neuronal está formada de un gran número de unidades de procesamiento no lineales interconectadas, también denominadas nodos o neuronas. La próxima ecuación representa la entrada neta a la unidad no lineal y está descrita como:

$$\text{net}_i = \sum_j w_{ij}x_j + b_i \quad (2.19)$$

y la figura muestra el modelo artificial de una neurona, el cual consiste de un combinador lineal seguido de una unidad no lineal.

El aprendizaje de una red neuronal consiste en hallar los valores precisos de los pesos de sus conexiones para que pueda resolver un determinado problema. El

proceso general consiste en ir introduciendo una serie de datos patrón y ajustar los pesos siguiendo un determinado criterio.

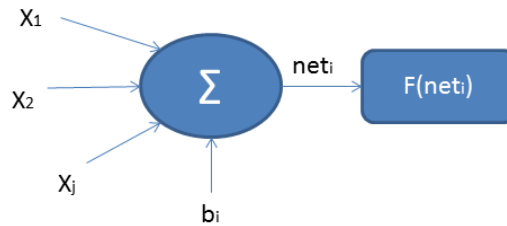


Figura 2.9: Red Neuronal Simple.

Los principales componentes de una red neuronal son:

- Un conjunto de unidades de procesamiento (neuronas).
- Y_i : Un estado de activación por cada unidad, que es equivalente a la salida de la unidad.
- W_{ij} : Pesos sinápticos, representan el grado de comunicación entre la neurona artificial j (pre-sináptica) y la neurona artificial i (post-sináptica). Pueden ser excitadores o inhibidores.
- Una regla de propagación, que integra la información proveniente de las distintas neuronas artificiales y proporciona el valor del potencial post-sináptico.
- función F La activación de una función, que determina el nuevo nivel de activación basándose en la entrada efectiva $X_i(t)$, vector de entrada de n -variables, y el actual estado de activación $Y_i(t)$.
- b_i : Una entrada externa (también conocido como sesgo) para cada unidad. Esta es independiente de la entrada, pero se modela como una entrada unitaria con peso b , funciona como entrada de polarización.
- Una regla de aprendizaje, que determina la forma de actualización de los pesos sinápticos de la red (aprendizaje).

Las neuronas reciben aportes de sus vecinos o de fuentes externas y utilizan esto para calcular una señal de salida que se propaga a otras unidades. Aparte de este tratamiento, una segunda tarea es el ajuste de los pesos. El sistema es inherentemente paralelo en el sentido de que muchas unidades se pueden llevar a cabo sus cálculos al mismo tiempo

Dentro de las ventajas de las redes neuronales artificiales se encuentran:

- Aprendizaje adaptativo: capacidad de aprender a realizar tareas basadas en un entrenamiento o una experiencia inicial.
- Auto-organización: una red neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.
- Generalización: facultad de las redes neuronales de responder apropiadamente cuando se les presentan datos o situaciones a los que no habían sido expuestas anteriormente.
- Tolerancia a fallos: la destrucción parcial de una red conduce a una degradación de su estructura; sin embargo, algunas capacidades de la red se pueden retener, incluso sufriendo gran daño. Con respecto a los datos, las redes neuronales pueden aprender a reconocer patrones con ruido, distorsionados o incompletos.
- Operación en tiempo real: los computadores neuronales pueden ser realizados en paralelo, y se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.
- Fácil inserción dentro de la tecnología existente: Se pueden obtener circuitos integrados especializados para redes neuronales que mejoran su capacidad en ciertas tareas. Ello facilita la integración modular en los sistemas existentes.

2.4.1. Clasificación De Redes Neuronales

Las arquitecturas neuronales pueden ser clasificadas atendiendo a distintos criterios:

- Según su estructura en capas:
 - Las redes monocapa son redes con una sola capa. Para unirse las neuronas crean conexiones laterales para conectar con otras neuronas de su capa. Las redes más representativas son la red de Hopfield (figura 2.10), la red BRAIN-STATE-IN-A-BOX o memoria asociativa y las maquinas estocásticas de Botzmann y Cauchy. Entre las redes neuronales monocapa, existen algunas que permiten que las neuronas tengan conexiones a si mismas y se denominan autorecurrentes. Las redes monocapa han sido ampliamente utilizada en circuitos eléctricos ya que debido a su topología, son adecuadas para ser implementadas mediante hardware, usando matrices de diodos que representan las conexiones de las neuronas.

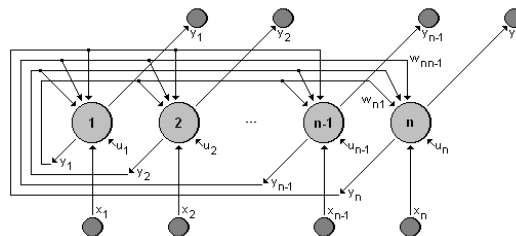


Figura 2.10: Red Hopfield.

- Redes multicapas, cuyas neuronas se organizan en varias capas (de entradas, oculta y de salida). La capa a la que pertenece la neurona puede distinguirse mediante la observación del origen de la señal que recibe y el destino de la misma.
- Según el Grado de Conexión:
 - Redes Neuronales Totalmente Conectadas: En este caso todas las neuronas de una capa se encuentran conectadas con las de la capa siguiente (redes no recurrentes) o con las de la anterior (redes recurrentes).

- **Redes Parcialmente Conectadas:** En este caso no se da la conexión total entre neuronas de diferentes capas. Estas estructuras neuronales se podrían conectar entre sí para dar lugar a estructuras mayores: estamos en el nivel de la mesoestructura. Esta conexión se puede llevar a cabo de diferentes formas siendo las más usuales, las estructuras en paralelo y jerárquicas. En la primera estructura se plantea un “consenso” entre las diferentes redes para obtener la salida mientras que en la estructura jerárquica existen redes subordinadas a otras que actúan como elementos centrales en la salida final de la red.

[16].

- Según la regla de aprendizaje: existen tres tipos básicos de paradigmas que pueden utilizarse para la actualización de los pesos sinápticos en una red neuronal, cuya distinción procede del campo de reconocimiento de patrones. La diferencia fundamental entre los tipos de aprendizaje radica en la existencia o no de un agente externo o supervisor que controla el proceso de aprendizaje de la red, estos tipos son:
 - **Aprendizaje supervisado:** se caracteriza por la presencia de un agente externo que controla el proceso de entrenamiento, estableciendo la respuesta que debería generar la red a partir de una entrada determinada. De esta forma, el supervisor comprueba la salida de la red, la compara con la deseada y, si existe diferencia, ajusta los pesos sinápticos hasta que la salida tienda a ser la deseada, utilizando para ello información detallada del error cometido en cada paso. En este tipo de entrenamiento figuran algoritmos como: aprendizaje por corrección de error (error de retropropagación o backpropagation), aprendizaje por refuerzo, aprendizaje estocástico, entre otros [17].
 - **Aprendizaje no supervisado:** este tipo de aprendizaje también se le llama auto-organizado y consiste en presentarle a la red un conjunto de patrones sin adjuntar la respuesta deseada, por lo que la red, por medio

del algoritmo de aprendizaje, desarrolla cualidades para detectar características de regularidades estadísticas en el universo de entrada. Ejemplo de este tipo de aprendizaje lo constituyen: el Aprendizaje hebbiano, formulado por Donald Hebb en 1949, y el Aprendizaje competitivo y cooperativo [18].

2.4.2. Tipos De Redes Neuronales

Las RNAs pueden ser distinguidas dentro de dos grandes categorías:

- Redes estáticas (Feedforward) fig. 2.11, las cuales no presentan conexiones retroalimentadas, ni retardos en los enlaces entre neuronas. La salida de la red se computa directamente a partir de la señal presente en la entrada y de las conexiones hacia delante de las diversas neuronas. Estas redes son adecuadas para el procesamiento de patrones espaciales. En este caso, la información relevante se distribuye a lo largo de las coordenadas espaciales asociadas con cada variable que componen el conjunto de entrada. El proceso de entrenamiento se lleva a cabo mediante el algoritmo de retropropagación (backpropagation) que es la generalización de la regla de aprendizaje Widrow-Hoff para redes de múltiples capas y funciones de transferencias no lineales [18].

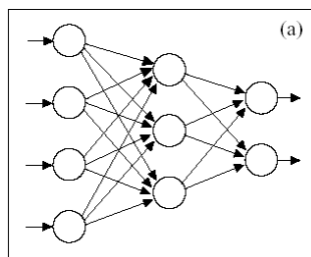


Figura 2.11: Red Feedforward.

- Redes neuronales artificiales dinámicas (RNAD o Feedback): se caracterizan por presentar conexiones retroalimentadas y/o retardos en los enlaces entre neuronas de la misma capa o de capas diferentes. La salida de este tipo de redes no solo depende del vector presente en la entrada, sino también de las

entradas anteriores o estados previos de la red. Por su característica inherente de memorizar la información pasada, durante lapsos de tiempo corto o largos, estas redes son buenas candidatas para procesar patrones con dependencia espacio-temporales, ejemplo de ello lo constituyen aplicaciones de procesamiento de señales con énfasis en la identificación y control de sistemas dinámicos no lineales y predicción de series de tiempo no lineales.

Las redes dinámicas son generalmente más potentes que las redes estáticas, aunque un poco más difíciles de entrenar. Los algoritmos de aprendizaje recurrente en tiempo real (RTRL, Real Time Recurrent Learning) y backpropagation a través del tiempo (BPTT, Backpropagation Through the Time) se basan en la propagación de errores a través de las conexiones recurrentes [19]. Esta estructura recurrente la hace especialmente adecuada para estudiar la dinámica de sistemas no lineales.

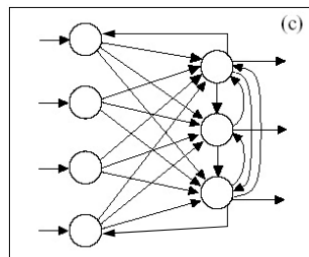


Figura 2.12: Red Hopfield.

- Redes Feedlateral: sistemas en donde las salidas de las neuronas de una capa, pueden ser entradas a las neuronas de la misma capa. (Aprendizaje Competitivo).

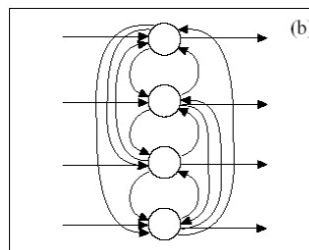


Figura 2.13: Red Feedlateral.

2.4.3. Tipos De Función De Activación

La función de activación $f(\text{net})$ define la salida de una neurona y existe una gran variedad de funciones que se han ido utilizando en las distintas bibliografías. Algunas de las típicas funciones de activación son:

- **Función Sigmoidal:** esta es la función de activación más común empleada en la construcción de Redes Neuronales MLP (Multicapas Perceptrón), la veremos más adelante.

$$f(\text{net}) = \frac{1}{1 + \exp^{-\frac{\text{net}}{\sigma^2}}} \quad (2.20)$$

- **Función Gaussiana:** esta es la función de activación mas común empleada en la construcción de Redes Neuronales RBF que también veremos a continuación.

$$\varphi = \exp\left(-\frac{1}{2\sigma^2} |x - c|^2\right) \quad (2.21)$$

c : Centro de la Gaussiana

σ^2 : Varianza

El esquema que vamos a utilizar para resolver nuestro problema de la determinación de la dirección de arribo de las señales incidentes se puede ver en la Figura 2.14. Que consiste de una cadena de baja conversión (complejo) y conversión análogo a digital (A / D). Debe haber M cadenas de baja conversión, una para cada uno de los elementos del arreglo. Una primera etapa de pre-procesado para transformar las señales que llegan a la agrupación de antenas en entradas para la red neuronal, dicho proceso fue descrito anteriormente en la sección 2.3. Una vez introducidas las entradas en la red neuronal obtenemos a la salida los ángulos de incidencia de las señales recibidas en la agrupación, pudiendo de esta forma detectar donde se encuentra un mayor número de usuarios, por ejemplo dentro de una celda de telefonía móvil.

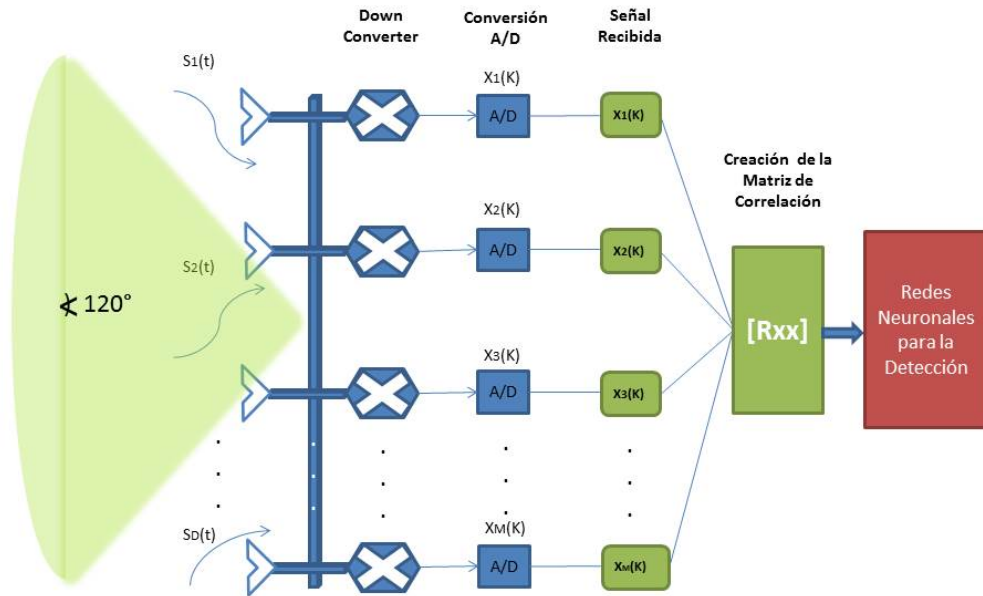


Figura 2.14: Esquema de la red neuronal para DOA.

2.4.4. Redes Perceptron Multicapa

El perceptrón multicapa con conexiones hacia adelante es una generalización del perceptrón simple. Surge como respuesta a los problemas que tenía dicha red, como por ejemplo, no poder resolver problemas que no fueran linealmente separables. De hecho, algunos autores han demostrado que el perceptrón multicapa es un aproximador universal de cualquier función continua en el espacio R^n .

2.4.4.1. Arquitectura del perceptrón multicapa

La arquitectura de este tipo de red se caracteriza porque tiene todas sus neuronas agrupadas en distintos niveles llamados capas. El primer nivel corresponde a la capa de entrada, que se encarga únicamente de propagar por el resto de la red

las entradas recibidas. El último nivel es el de la capa de salida. Se encarga de proporcionar los valores de salida de la red. En las capas intermedias denominadas capas ocultas, se realiza un procesamiento no lineal de los patrones recibidos. Las conexiones del perceptrón multicapa son hacia adelante. Generalmente todas las neuronas de un nivel se conectan con todas las neuronas de la capa inmediatamente posterior. A veces, dependiendo de la red, se encuentran conexiones de neuronas que no están en niveles consecutivos, o alguna de las conexiones entre dos neuronas de niveles consecutivos no existe, es decir, el peso asociado a dicha conexión es constante e igual a cero. Además, todas las neuronas de la red tienen un valor umbral asociado. Se suele tratar como una entrada cuyo valor es constante e igual a uno, y lo único que varía es el peso asociado a dicha conexión (que es el umbral realmente).

Por otro lado, las funciones de activación que se suelen utilizar son la función identidad, la función sigmoideal y la función tangente hiperbólica.

El algoritmo retropropagación es el algoritmo básico de aprendizaje que usa el perceptrón multicapa, por ende, es el que se utilizara. Es el algoritmo mediante el cual se van adaptando todos los parámetros de la red. El tipo de entrenamiento que sigue este tipo de red es supervisado. El aprendizaje de la red se plantea como un problema de minimización de una determinada función de error. En general se usa como función del error, el error medio cuadrático, es decir:

$$E = \frac{1}{N} \sum_{n=1}^N e(n) \quad (2.22)$$

Dónde:

$$e(n) = \frac{1}{2} \sum_{n=1}^t (s(n) - y(n))^2 \quad (2.23)$$

Siendo $s(n)$ es la salida patrón, $y(n)$ la salida obtenida de la red, t el número de neuronas de salida y N el número de patrones [20].

2.4.5. Redes neuronales de base radial

Las redes neuronales de base radial surgen con el objeto de conseguir redes neuronales cuyo proceso de aprendizaje fuera mucho más rápido que en las redes perceptrón multicapa. Para ello se diseñó una red en la que las neuronas de su única capa oculta tuvieran un carácter local, de tal manera que a la entrada de nuevos patrones hubiera menos neuronas que procesar. Ésta es de hecho su principal diferencia con las redes perceptrón multicapa, el comportamiento local de sus neuronas, de tal manera que solo unas pocas se activan ante la entrada de un determinado patrón. Esto le confiere determinadas ventajas e inconvenientes en comparación con las primeras.

De todas maneras, y al igual que sucedía con el perceptrón multicapa, es un aproximador universal en el espacio $\mathcal{R}^n$ [20].

2.4.5.1. Arquitectura de Redes neuronales de base radial

Las redes neuronales de base radial disponen de una capa de neuronas de entrada, otra capa de neuronas de salida y solo una capa de neuronas ocultas. Las conexiones en este tipo de red son todas hacia adelante, es decir, la entrada de todas las neuronas de la capa oculta procede de la salida de todas las neuronas de la capa de entrada, y a su vez las neuronas de la capa oculta transmiten su salida a todas las neuronas de la capa de salida. Un ejemplo de arquitectura de este tipo de red se describe en la subsección 2.4.2 específicamente las redes feedforward.

Según lo que hemos dicho de cómo se propaga la señal a través de la red la salida Y_i bajo un determinado patrón de entrada $X = (x_1 x_2 x_3 x_i)$, donde x_i es la entrada a la neurona i , de esta manera para este tipo específico de red neuronal, su ecuación viene dada por:

$$y_1 = \sum_{i=1}^N w_{1i} \phi_i(X) + b \quad (2.24)$$

La función $\phi(X)$ es la función de activación de la neurona i de la capa oculta. Las neuronas de la capa oculta tienen un comportamiento local debido al tipo de función que es. Esta función tiene la forma siguiente:

$$\phi_i(X) = \phi\left(\frac{|X - C_i|}{d_i}\right) \quad (2.25)$$

Es decir, el nivel de activación de la neurona i de la capa oculta depende de la distancia entre el patrón y el centro C_i asociado a dicha neurona. El parámetro d_i mide de alguna forma el campo de actuación de la neurona. De esta manera, si la entrada se encuentra cerca del centro de la neurona el nivel de activación de la misma será grande, y en caso contrario pequeño. Podemos observar como el centro C_i determina la zona de actuación de la neurona y el parámetro d_i la extensión de dicha zona. Por ello se dice que estas neuronas tienen un comportamiento local. Esta función de activación, también conocida como función de base radial puede adoptar diversas formas que se adecuan a este esquema. Algunas de ellas son:

Función gaussiana:

$$\phi_r = \exp\left(\frac{-r^2}{2}\right) \quad (2.26)$$

Función inversa cuadrática:

$$\phi_r = \frac{1}{1 + r^2} \quad (2.27)$$

Función inversa multicuadrática:

$$\phi_r = \frac{1}{\sqrt{1 + r^2}} \quad (2.28)$$

Donde $r = \frac{|X - C_i|}{d_i}$, es decir, la distancia euclídea entre la entrada A y el centro de la neurona dividido por la amplitud. Hay que tener en cuenta que el centro

tendrá tantas componentes como entradas tenga la red para situarlo en el espacio R^N , donde N es el número de neuronas de entrada que existen en la red [20].

Hay que determinar, dependiendo de la arquitectura de la red la cantidad de neuronas en la capa de entrada que viene dado por la cantidad de grados a detectar, es decir, el número de neuronas en la capa de entrada viene determinado por las entradas necesarias para la resolución del problema; la capa de salida sería el sensor de activación de los grados detectados, el cual es un valor variable. La determinación del número óptimo de neuronas en la capa oculta es más difícil de determinar, ya que, hay que tener en cuenta que si existen pocas neuronas, puede haber errores en la solución del espacio del problema, esto se refiere a la tolerancia del sensor detector, a la amplitud de detección, y por el contrario, si existen exceso de neuronas, pueden existir varias neuronas en una determinada región, lo que conlleva consecuencias negativas en la aptitud de la red para resolver el problema, ya que podría solo determinar la detección para un criterio específico y no amplio como se desea. Por esto se utiliza el procedimiento ensayo error añadiendo cada vez más neuronas hasta obtener un resultado óptimo, este proceso lo realiza el software automáticamente de manera que el aprendizaje sea no supervisado.

Capítulo III

Procedimientos de la investigación

3.1. Metodo a Desarrollar

Con el objeto de desarrollar redes neuronales las cuales sean capaces de detectar la dirección de arribo de señales incidentes a un arreglo de antenas, se decidió, manejar redes las cuales, hayan demostrado ser lo suficientemente versátiles como para manejar este tipo de información, de esta manera se escogió dos tipos de red, especializadas en manejo de datos no lineales, siendo la no linealidad la característica fundamental en la detección de la dirección de arribo. Los dos tipos de red neuronal escogido fueron:

- Red Neuronal de Base Radial (RBF)
- Red Neuronal Perceptrón Multicapa (MLP)

Para lograr una adecuada caracterización de las neuronas en el proceso de entrenamiento, se planteó un proceso de detección especificada en la figura [3.1](#)

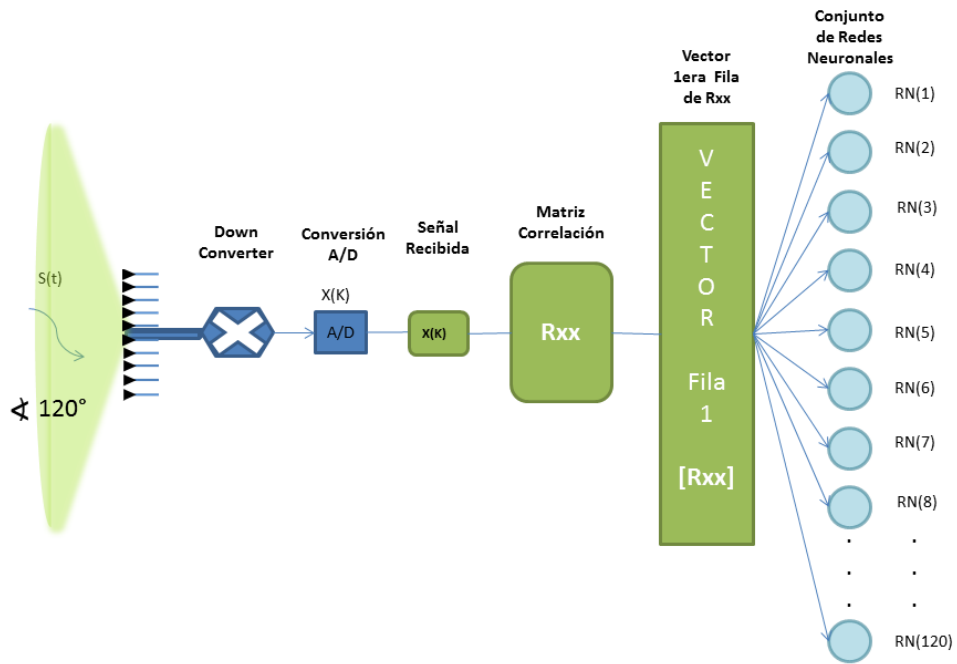


Figura 3.1: Esquematización.

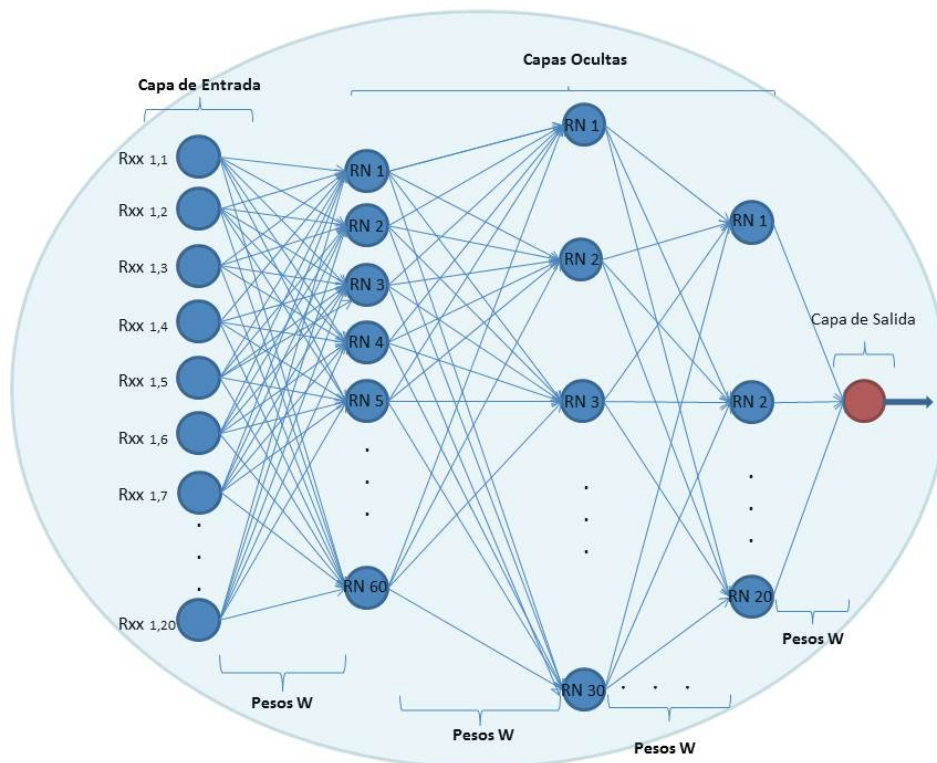


Figura 3.2: Composición de la Red Neuronal MLP.

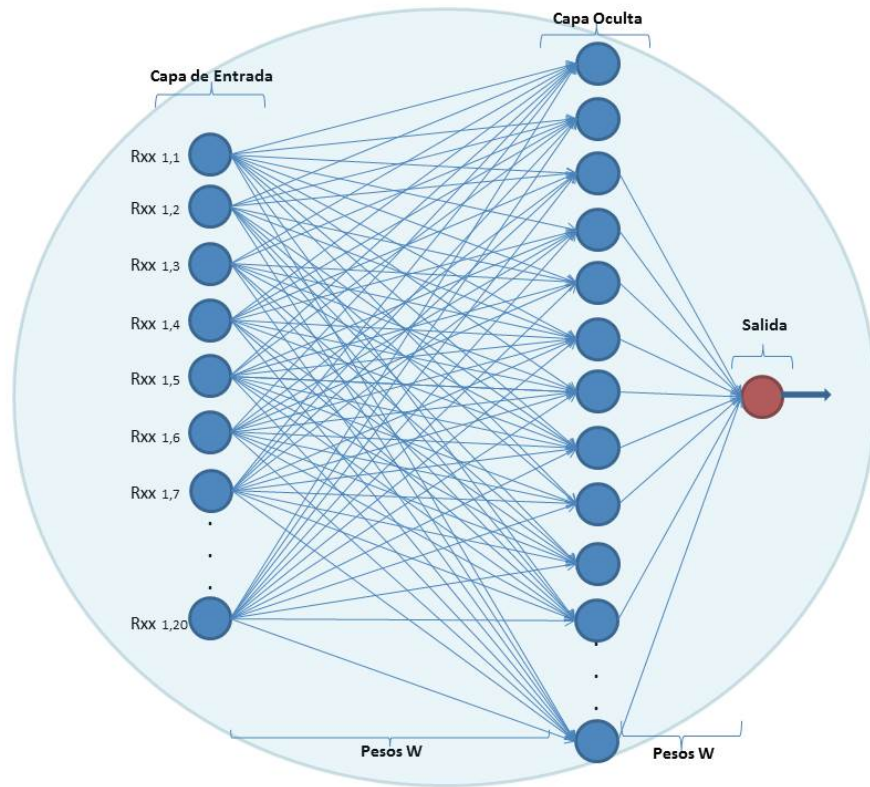


Figura 3.3: Composición de la Red Neuronal RBF.

Como se observa claramente en la imagen, el planteamiento es generar una red neuronal por **cada grado**, la cual se encargara de indicar si en ese grado especifico, se encuentra o no una señal de llegada.

Para otorgarle un carácter formal, practico y real, las señales de llegada serán presentadas como señales de telefonía celular móvil GSM, las cuales para este caso específico, se tomara como frecuencia de portadora 900 MHz.

Otra de las características tomadas del campo laboral, en donde se espera esta tesis pueda aportar algún adelanto, indica que las antenas inteligentes de telefonía abarcan en su configuración un campo de 120° , de esta manera con el uso de 3 juegos de antenas se pueden cubrir perfectamente 360° . [3.4](#)

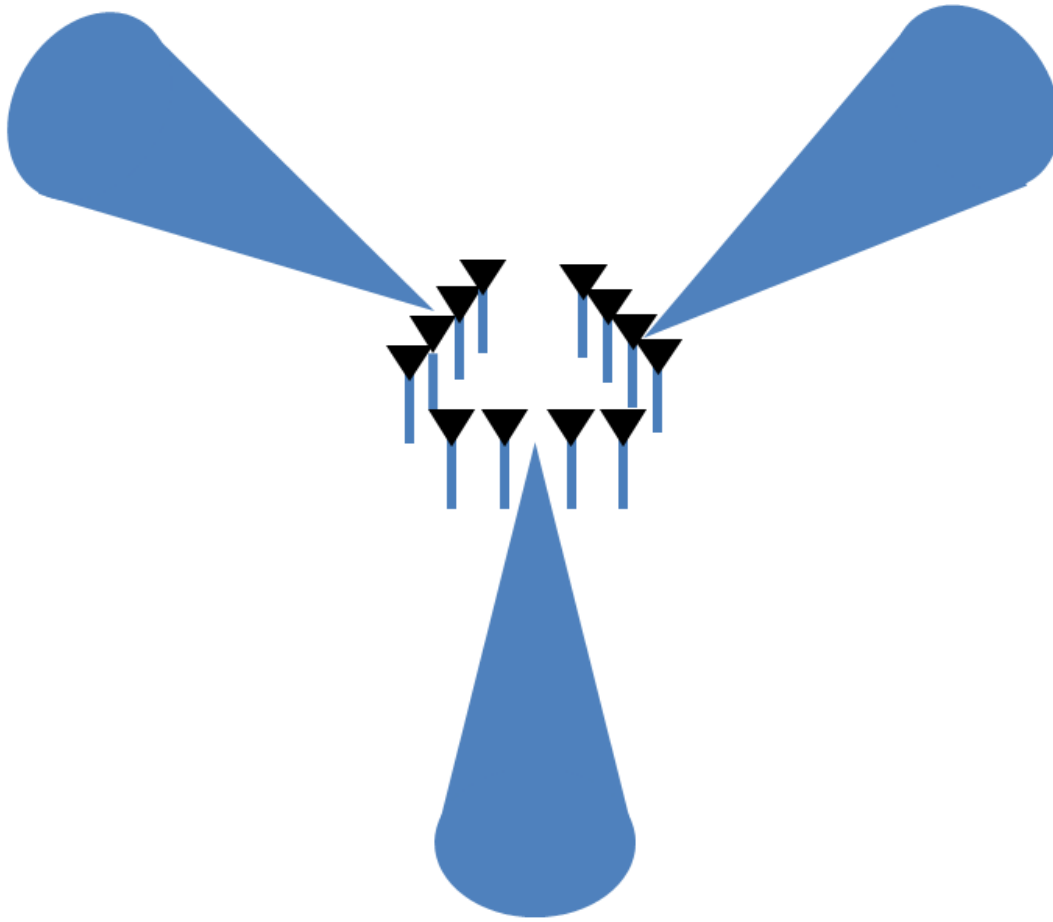


Figura 3.4: Sistema de Detección cubriendo 360°.

3.1.1. Creación de las Redes Neuronales

Para poder realizar de manera óptima y eficiente la creación de las distintas redes neuronales se usó el programa MATLAB, dado que tiene herramientas útiles y diversas en el desarrollo de redes neuronales.

Este programa trae por defecto ciertos comandos, los cuales facilitan al usuario el desarrollo de las redes neuronales, en el caso de las redes de base radial RBF se usó el comando `newrb`.

En este comando se configuran ciertos parámetros como los son: `P`, `T`, `goal`, `spread`, `MN`, `DF`.

- *P*: es la matriz de entrenamiento de entrada de la red neuronal, esta matriz está conformada por los distintos vectores normalizados de llegada de cada uno de los posibles ángulos de arribo.
- *T*: representa el "target", es decir la salida deseada de la red neuronal en proceso de entrenamiento.
- *goal*: es el error medio cuadrático (MSE) que se desea obtener de la red neuronal en proceso de entrenamiento, para el caso de esta tesis se usó como límite $MSE=0.01$.
- *spread*: indica el radio de la función radial a usar en el entrenamiento de la red RBF, luego de numerosos ensayos de llego a demostrar que el mejor rendimiento se obtiene con un $spread=0.6$
- *MN*: establece el número máximo de neuronas a usar, en este caso se estableció igual al número de vectores de entrada.
- *DF*: indica cada cuantas iteraciones se gráfica el rendimiento de la red a entrenar, parámetro fijado en 25, para seguir el proceso de aprendizaje de la red.

Es importante señalar que con cada iteración de la red RBF en el proceso de aprendizaje, se crea una neurona más en su capa oculta hasta obtener el mínimo MSE estipulado, el promedio de neuronas en la capa oculta fue de 300 neuronas para lograr el mínimo error estipulado.

Por otra parte en el proceso de elaboración de las redes neuronales MLP, se usó el comando `newff`, con el comando se logra un manejo simple de las redes y se logra un control sencillo del proceso de aprendizaje.

Este comando tiene ciertos parámetros que deben introducirse para su correcto funcionamiento como lo son: *P*, *T* y *S_i*, donde *P* y *T* cumplen la misma función que su análogo en redes RBF, y *S_i* es el tamaño en neuronas de cada *i*-ésima capa de la red MLP.

Para entrenar este tipo de redes se usa el comando `train`, el cual solo necesita los parámetros *P*, *T* y la red creada con el comando `newff`.

Cada red creada bajo este criterio tiene ciertas características iguales entre sí como lo son:

- La función de transferencia de cada neurona presente en cada capa oculta es la tangente hiperbólica, excepto en la última neurona (neurona de salida), donde se usa una función lineal.
- la función de entrenamiento es `newff` La cual además de ser de las más rápidas en llegar a la convergencia, ha dado magníficos resultados en la creación de las redes.
- Como función de aprendizaje para obtener el menor gradiente se usa `learngdm`

Las redes neuronales de tipo MLP, se generaron siguiendo un patrón de Backpropagation como se muestra en la imagen 3.5, el cual ha demostrado su eficiencia en situaciones de no linealidad [15], esta eficiencia se mide por su MSE que al igual que en el caso de las RBF, puede ser un valor que el usuario introduzca.

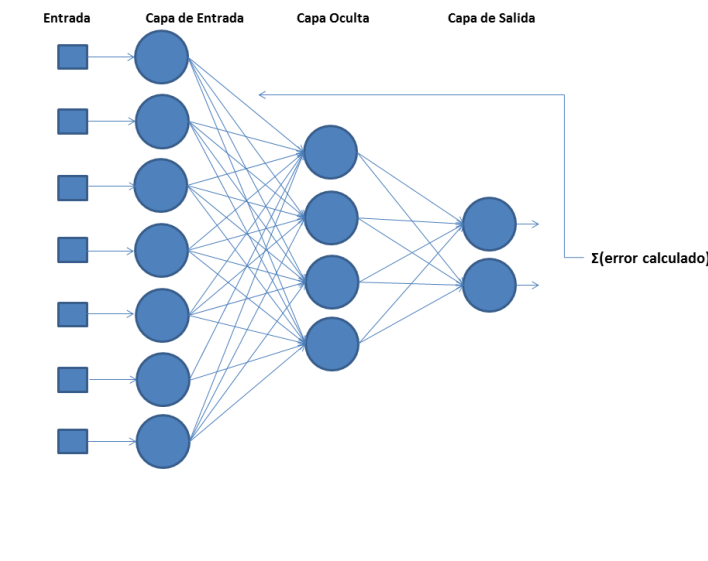


Figura 3.5: Red MLP Backpropagation.

El proceso de creación y posterior entrenamiento de las redes neuronales artificiales más que una ciencia es un verdadero arte, esta afirmación se acentúa al trabajar con redes de tipo MLP debido al que el diseño de las distintas capas y el número

de neuronas que cada una de esas capas debe tener es simplemente producto de la experimentación, o ensayo y error, luego de un número considerablemente alto de mediciones y múltiples sondeos, se dedujo que la mejor estructura de capas ocultas y el tamaño de estas era 3 capas ocultas para cada red neuronal a usar y las capas ocultas serían de 60 neuronas para la primera capa, 40 neuronas para la segunda, y 30 para la tercera. Sumado a esto tienen una capa de entrada de 20 neuronas y una capa de salida de una sola neurona.

3.2. Caracterización de los Vectores de Entrenamiento

El pilar fundamental que define a una red neuronal bien entrenada, es la información o datos que se usen para su entrenamiento, estos datos no solo deben ser estructurados de la manera correcta, sino que también deben representar de manera fehaciente las características de los elementos a representar.

Existen ciertos parámetros que se tomaron como fijos debido a que en la realidad, estos parámetros son inamovibles, entre ellos se encuentran el número de dipolos que se usa en la recepción de los datos, y la distancia existente entre ellos, como se ilustra en la imagen 3.6, número conveniente, guardando una relación excelente entre capacidad de recepción y costo computacional.

La distancia entre estos dipolos, se fijó en $\lambda/2$ debido a que es el valor más conveniente, como se explicó en la sección 2.2.2.

Una vez establecido estos parámetros, se procede a la formación de las matrices tanto de señales recibidas como de vectores de dirección de llegada, explicados en la sección 2.3 del capítulo II.

Como ya se mencionó en la sección 3.1 las señales que arribaran al arreglo, son de estructura GSM y tienen la forma:

$$s(t) = \sin(2\pi \times f_b \times t) \sin(2\pi \times f_p \times t) \quad (3.1)$$

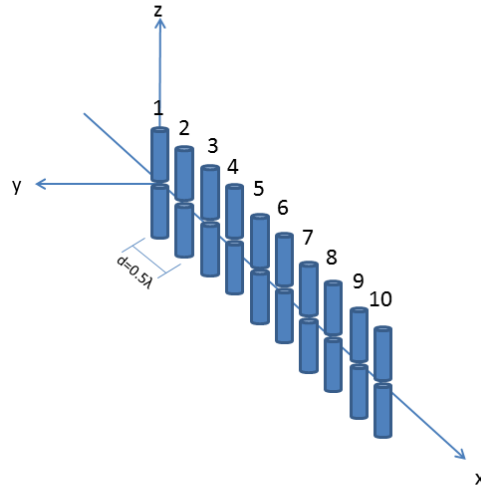


Figura 3.6: Arreglo de 10 dipolos con distancia de media longitud de onda entre ellos.

Como se puede ver en la ecuación 3.1 la información se tomó como una onda monocromática con una frecuencia $f_b = 10\text{KHz}$, mientras que la onda portadora, tiene una frecuencia f_p , la cual está en el orden de los 900MHz .

Para el caso específico de esta investigación y sabiendo que la frecuencia de subida barca desde 890MHz a 915MHz , se estableció que las frecuencias de uplink, fueran a partir de 900MHz con incrementos de 200KHz , hasta alcanzar los 124 canales del standar. La frecuencia portadora fue multiplexada a 50MHz , y se muestreo a 200MHz , para emular de manera fehaciente el proceso real que es llevado a cabo en los equipos de antenas móviles.

Luego se dispuso de la matriz de direccionamiento 2.19, tal como se expuso en la sección 2.3.1.

Generalmente los algoritmos que trabajan con agrupaciones de antenas utilizan la matriz de correlación para la estimación de dirección de llegada en lugar de la

señal $X(t)$ 2.8, ya que la correlación nos aporta mucha más información sobre las señales recibidas. El vector de entrada a la red neuronal son los elementos de la matriz de correlación R_{xx} que se organizan como un solo vector b de longitud $2M$ ya que la red no trabaja directamente con números complejos y debemos separar las partes reales e imaginarias.

El vector de entrada b es dividido por su norma para asegurar que la red trabaje con un rango de valores normalizados de entre 0 y 1 . Finalmente, los valores del vector z constituyen la entrada a la red neuronal.

$$z = \frac{b}{\|b\|} \quad (3.2)$$

Para explotar la simetría en la matriz de correlación R_{xx} solo se necesita considerar la parte triangular superior o inferior de la matriz, pero se ha demostrado que la información necesaria para garantizar un correcto aprendizaje se encuentra en la primera fila de la matriz de correlación lo cual simplifica aún más el cálculo.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (3.3)$$

$$b = \begin{bmatrix} r_{11} & r_{12} & r_{13} \end{bmatrix} \quad (3.4)$$

Hay que mencionar que el entrenamiento de una red neuronal para detectar el ángulo de llegada de múltiples fuentes no es una tarea fácil. Para tener una idea de cuánto entrenamiento es requerido consideremos el problema de seguir a dos fuentes solamente. Primero consideramos el espacio en la agrupación desde -60 a 60 y empezamos el entrenamiento con dos señales θ (señal 1) y ϕ (señal 2), una en el ángulo $\theta = -60$ y la otra en $\phi = -59$, luego se mueve un poco la señal 1 y se coloca $\theta = -59$ $\phi = -59$, luego $\theta = -58$ $\phi = -59$, y así sucesivamente hasta tener $\theta = 60$ $\phi = -59$, seguidamente se mueve un poco la señal 2 a $\phi = -58$ y se comienza todo

el ciclo nuevamente, el entrenamiento se considera completo cuando ambas señales han recorrido todos los ángulos disponibles.

A continuación se enumera un resumen de los pasos a seguir para la construcción de los datos de entrada al entrenamiento de las redes neuronales:

1. Construcción de la señal $X(t) = A \cdot S(t) + N(t)$.
2. Calculo de la matriz de correlación R_{xx} .
3. Formación el vector b .
4. Obtenemos el vector z normalizando b .
5. Generación final de los pares de entrada y salida.

La salida, estará conformada por un vector cuya longitud viene determinada por el número de posibles llegadas que pueda tener la red, este vector estará conformado únicamente por ceros en aquellas circunstancias no favorables y por amplitudes de 3 en las circunstancias favorables (que la entrada corresponda al ángulo que se está entrenando), se planteó realizarlo con amplitudes de 3 debido a las buenas prestaciones arrojadas en las diversas experimentaciones realizadas, entre estas se pueden mencionar, colocar amplitud 1, cuando se recibe una señal correcta, y 0 para cuando se recibe una señal que no se encuentra en el grado especificado, este tipo de configuración fallo al no establecer suficiente diferencia entre una señal correcta y otra que no, sobre todo al emplear mas de 2 señales de arribo, es notorio la poca resolución de este método.

otro de los esquemas planteados fue el de un valor negativo para una señal que no este en el angulo esperado, y un valor positivo para una señal que si este en el angulo esperado, este tipo de salida, igualmente no produjo los resultados esperados, debido a que nunca se estableció un error medio cuadrático menor a 0.1, mismo resultado que al intentar con una amplitud de 0 a la salida de la neurona en caso de no ser el angulo esperado, y una amplitud mayor a 3 cuando la señal de llegada si sea el ángulo esperado. Por estos motivos se estableció particularmente esa salida como respuesta en las neuronas.

Se espera que con este entrenamiento cada neurona pueda distinguir cuando una señal proviene exactamente del ángulo para el cual se entrenó y para ellos se debe entrenar con todas las direcciones y combinaciones posibles de ángulos de llegada.

3.3. Programas de Entrenamiento y Evaluación

Para agilizar el proceso de entrenamiento y posterior evaluación de las redes creadas, se desarrollaron 3 scripts en MATLAB, los cuales se encuentran en el apéndice A a continuación se muestran los diagrama de flujo de las funciones creadas. 3.7 muestra el proceso por el cual las redes neuronales fueron entrenadas.

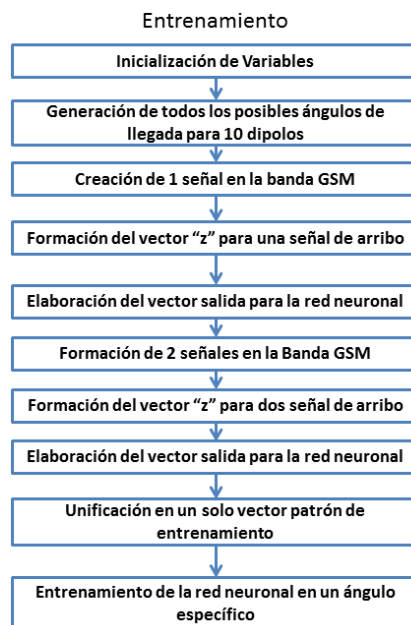


Figura 3.7: Diagrama de los pasos usados para entrenar cada red

La figura 3.8 bosqueja de qué manera se obtuvo el rendimiento de cada red creada.

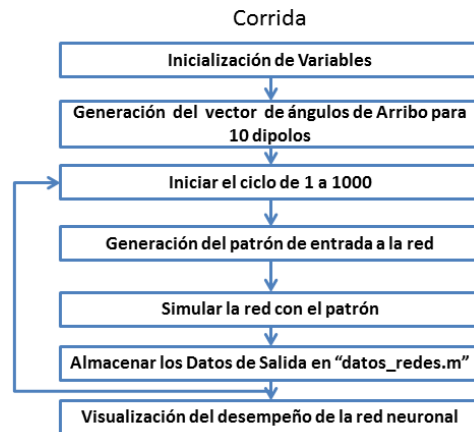


Figura 3.8: Diagrama de los pasos usados para evaluar cada red

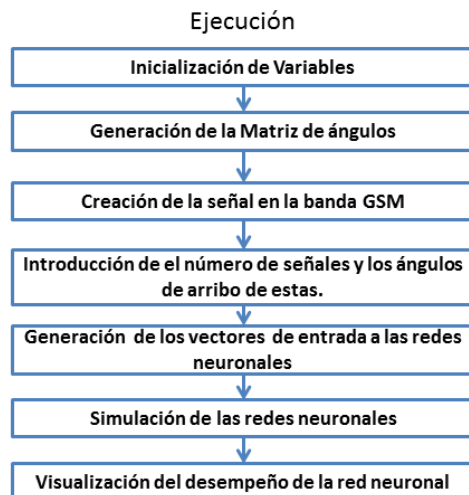


Figura 3.9: Diagrama del programa usado detectar la dirección de arribo

Es conveniente señalar que estas redes fueron creadas en computadores de cuyos procesadores son de 2,53GHz y 1,8GHz, y una memoria RAM de 2GB y 3GB respectivamente, la versión de MATLAB usado es el 2012b.

Es inherente de todo programa tener limitaciones, en cuanto a su capacidad de cálculo, en este caso no fue la excepción, se trató por todos los medios de que los datos de aprendizaje fuesen los más completos posibles pero debido al uso excesivo de memoria RAM, en el proceso de entrenamiento de las redes neuronales, se tuvo

que establecer un límite de entrenamiento de hasta 2 redes neuronales de entrada, esto es porque para sola señal de llegada es necesario generar una matriz de 10 filas (una por cada dipolo) por 121 columnas (una por cada ángulo de arribo), para 2 señales de arribo en caso de que se quiera cubrir los 121 grados tomando en cuenta cualquier combinación de ambas produciría una matriz de 10 filas por 14653 columnas, con lo cual aunado con los diferentes procesos y variables auxiliares que se deben generar para entrenar a las neuronas, desborda la capacidad de los equipos usados, por ende se estableció tener una resolución de 1 grado para una señal de llegada y de 2 grados para 2 señales de llegada con lo cual la matriz de entrenamiento se reduce a 10 filas por 3842 columnas sumando los casos de una sola señal y dos señales de arribo.

3.3.1. Performance de las Redes Generadas

Una vez creadas las redes, se ejecuto el código *ejecucion* para visualizar su respuesta para 1 y 2 señales de arribo. Primeramente se evaluó, la detección de arribo para una señal, usando el conjunto de redes neuronales RBF (figura 3.10)

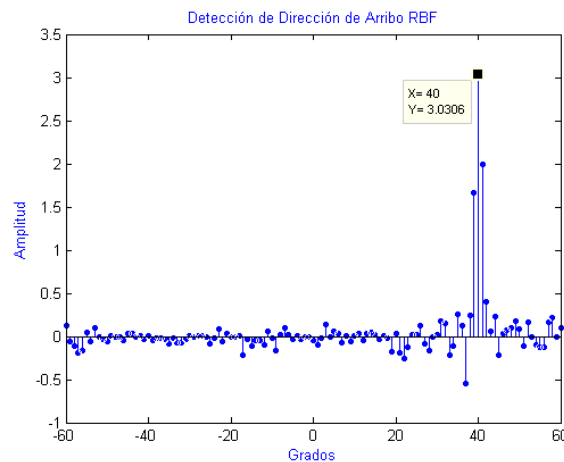


Figura 3.10: Detección de una señal con un ángulo de arribo de 15°

Seguidamente se muestra como es recepción de una sola señal, pero usando MLP en 3.11

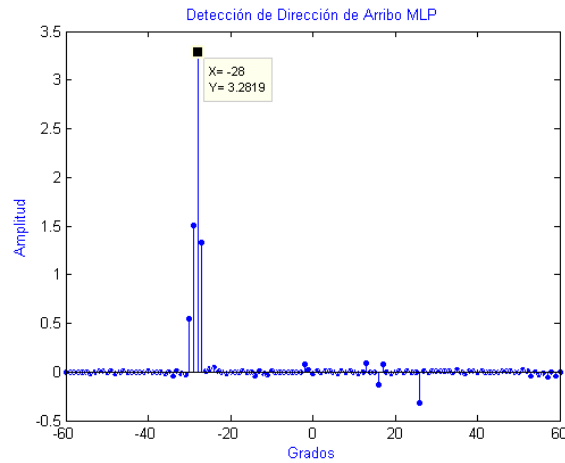


Figura 3.11: Detección de una señal con un ángulo de arribo de 6°

La detección de 2 señales incidentes usando RBF se muestra en [3.12](#)

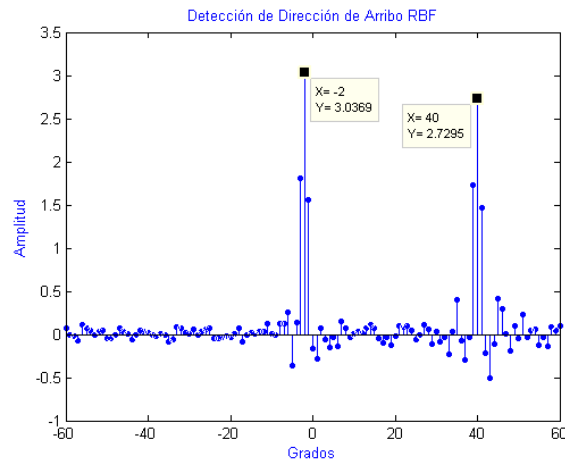


Figura 3.12: Detección de 2 señales con un angulo de arribo de -20° y 37°

La detección de 2 señales incidentes usando MLP se muestra en [3.13](#)

3.3.2. Algoritmos MUSIC y DML

Al ser estos los dos algoritmos más usados a nivel mundial para evaluar la dirección de arribo de señales inalámbricas, existe un gran compendio de información

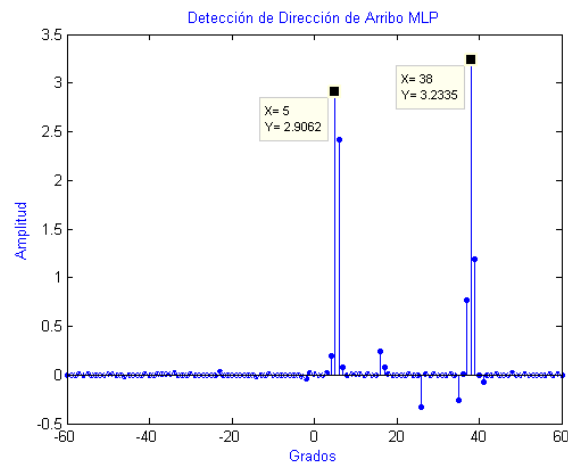


Figura 3.13: Detección de 2 señales con un ángulo de arribo de -35° y 20°

acerca de cómo generar estos algoritmos con códigos computacionales.

Se tomó como referencias [21] y [6], en el proceso de elaboración de los códigos, los cuales se encuentran en el apéndice B.

Capítulo IV

Análisis, interpretación y presentación de los resultados

4.1. Estructura para comprensión de resultados

Una vez realizadas todas las compilaciones se creó una base para presentar los resultados, y facilitar una mayor organización y comprensión de los mismos. A continuación se hará presentación de los resultados obtenidos según el orden siguiente:

- Ideal: En condiciones óptimas
- SNR=20: con relación señal a ruido de 20dB (condiciones reales)
- SNR=5: con relación señal a ruido de 5dB (condiciones no favorables)

Para cada caso se mostrarán las gráficas obtenidas para las distintas formas del cálculo de dirección de arribo, como son redes neuronales (RBF y MLP), y algoritmos (MUSIC y DML), para los distintos casos:

- 1 señal con ángulo de arribo θ_i

- 2 señales con ángulo de arribo θ_i y θ_j separadas.
- 2 señales con ángulo de arribo θ_i y θ_j considerablemente juntas.
- 2 señales con ángulo de arribo θ_i y θ_j próximas entre si.
- 4 señales con ángulo de arribo $\theta_1, \theta_2, \dots, \theta_4$.
- 7 señales con ángulo de arribo $\theta_1, \theta_2, \dots, \theta_7$.

Para ($i=1, 2, 3\dots$) y ($j=1, 2, 3\dots$)

De esta manera se realizara una comparación entre las formas de detección de arribo en cada caso, indicando la interpretación de los resultados y el análisis comparativo.

Es importante resaltar que en las gráficas, el eje de las abscisas representa los grados de incidencia que abarcan de -60° a 60° , y el eje de las ordenas solo representa una amplitud de detección en veces, esto para el caso de las redes neuronales y MUSIC; en DML en el número de iteraciones se representa en el eje de las ordenadas.

Tabla 4.1: Variables

	Nomenclatura	Valor
Frecuencia	F	900 Mhz
Separación entre canales	ΔF	200 Khz
Frecuencia Intermedia	FI	50 MHz
Número de señales incidentes	D	[1; 7]
Número de Dipolos	M	10
Ángulo de incidencia	θ	[-60; 60]
Distancia entre dipolos	d	$0,5\lambda$
Relación Señal a ruido	SNR	[60, 20, 5, -10] dB
Número de muestras	k	1000

4.2. Resultados en condiciones óptimas

4.2.1. 1 señal Incidente

Se presentarán los resultados para cada tipo de simulación empleada para la detección de arribo, únicamente mostrando una señal incidente de ángulo negativo, y posteriormente de ángulo positivo, en el caso donde no hay influencia del ruido en la señal.

θ	
-15°	46°

Redes Perceptrón Multicapa

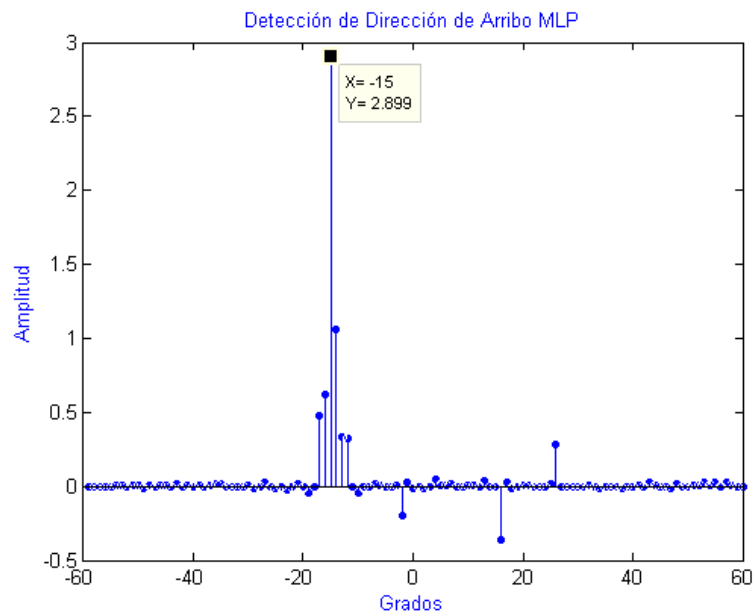


Figura 4.1: MLP $\theta = -15$.

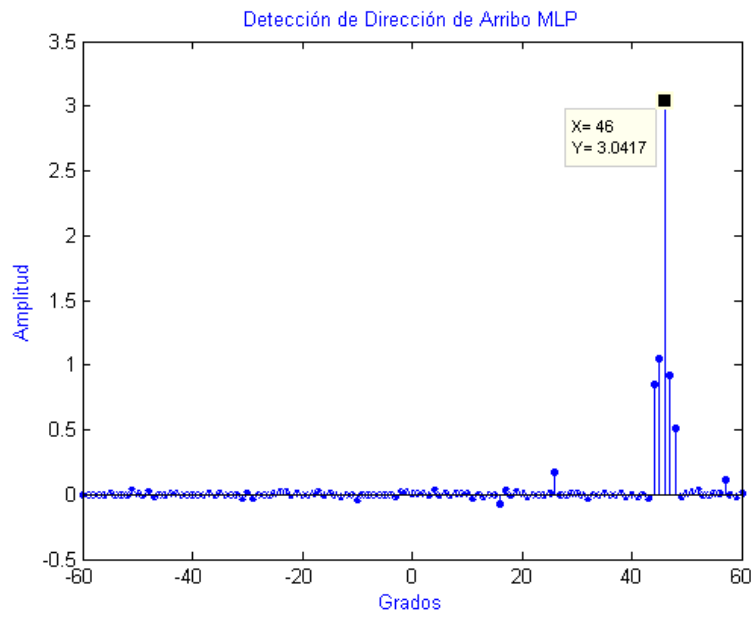


Figura 4.2: MLP $\theta = 46$.

Redes Neuronales de Base Radial

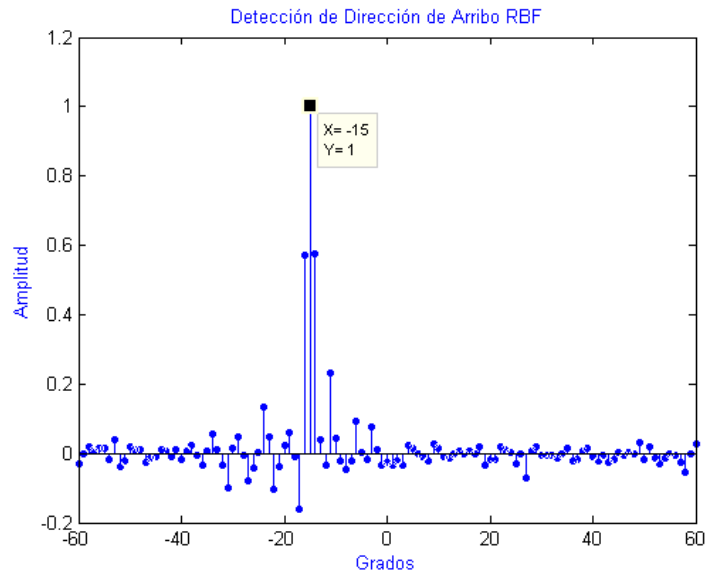
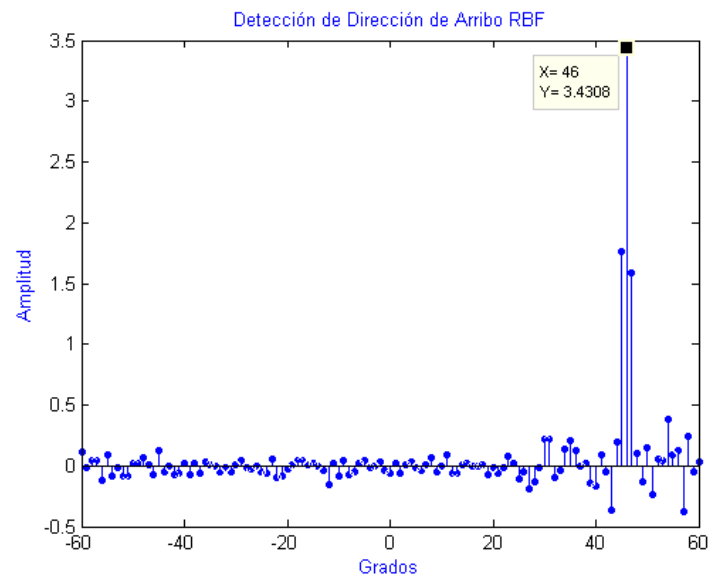
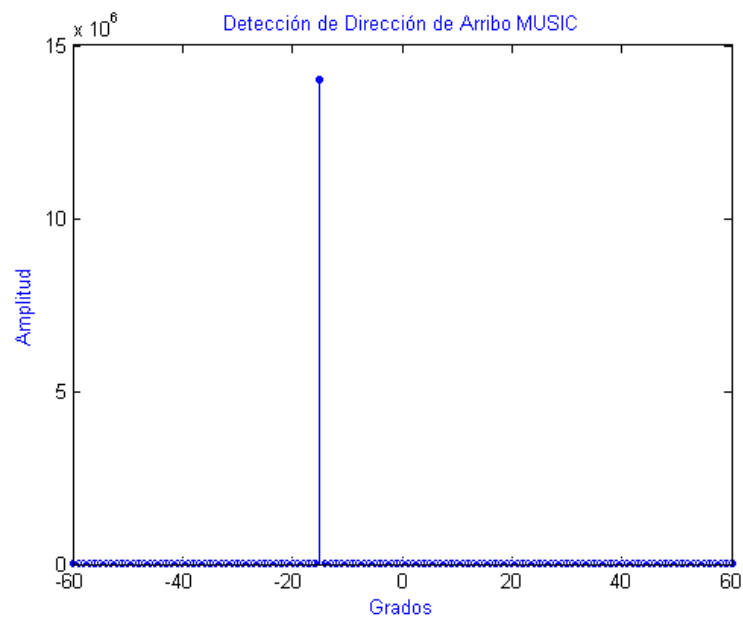
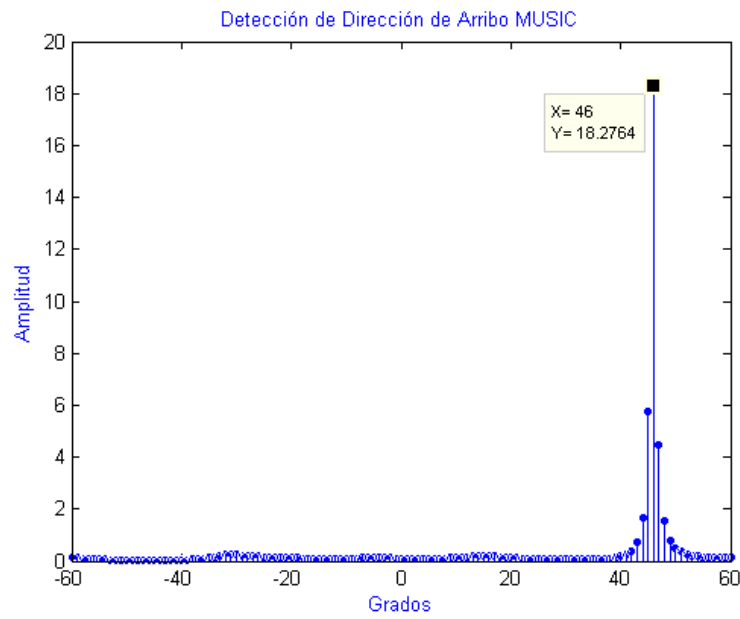


Figura 4.3: RBF $\theta = -15$.

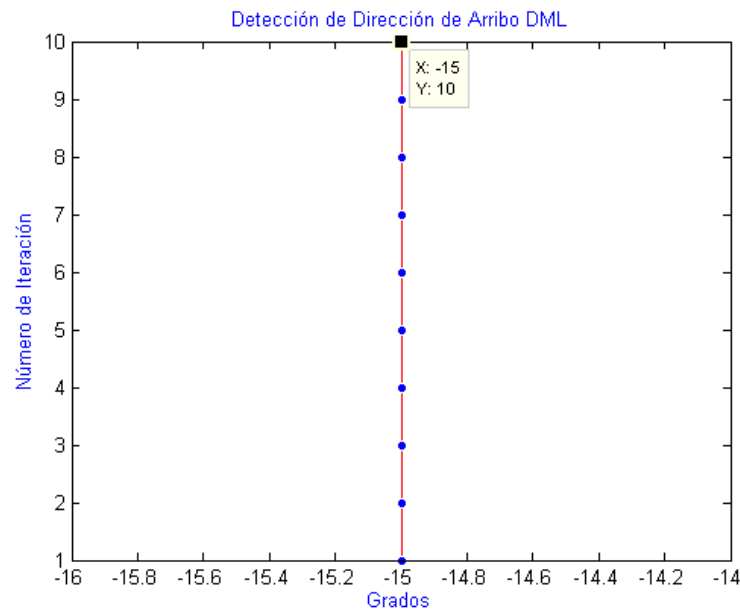
**Figura 4.4:** RBF $\theta = 46$.

Multiple Signal Classification

**Figura 4.5:** MUSIC $\theta = -15$.

Figura 4.6: MUSIC $\theta = 46$.

Deterministic Maximum Likelihood

Figura 4.7: DML $\theta = -15$.

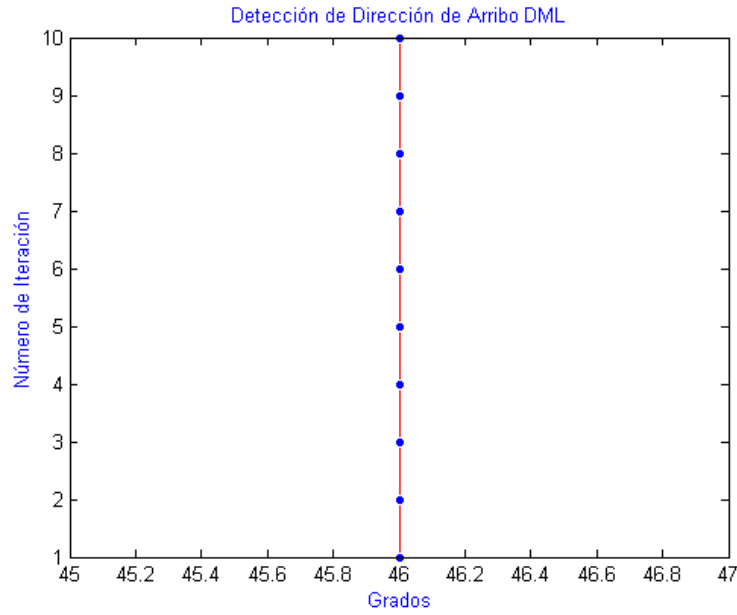


Figura 4.8: DML $\theta = 46$.

De todo lo observado se puede notar como las detecciones cumplen lo establecido, dado que detectan el mayor pico en amplitud para cada señal de -15° y 46° por cada gráfico distinto, existen mínimas variaciones en las redes neuronales (4.1, 4.2, 4.3, 4.4) cercanas al ángulo de detección, pero son de baja sensibilidad como para ser detectadas con respecto al ángulo de arribo θ , DML otorga un buen resultado en las 10 iteraciones establecidas (4.7, 4.8), por su parte MUSIC arroja resultados precisos igualmente (4.6, 4.5), por lo que se puede enfatizar que en condiciones ideales los algoritmos tienen mayor exactitud en determinación del ángulo de arribo, aunque no se descarta que las redes neuronales cumplen el mismo objetivo.

4.2.2. 2 señales Incidentes separadas

Una vez verificado para una señal, se ejecuta el comando del software para hacer los cálculos respectivos con dos señales incidentes y así observamos su comportamiento mientras estas señales estén separadas con respecto al grado de incidencia, teniendo presente que nos encontramos en condiciones ideales.

θ_1	θ_2
-37°	48°

Redes Perceptrón Multicapa

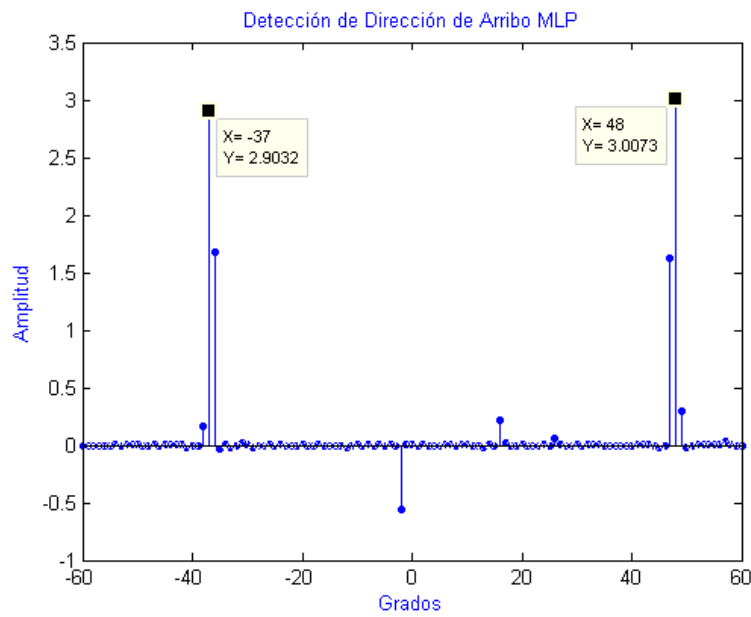


Figura 4.9: MLP $\theta_1 = -37$ y $\theta_2 = 48$.

Redes neuronales de base radial

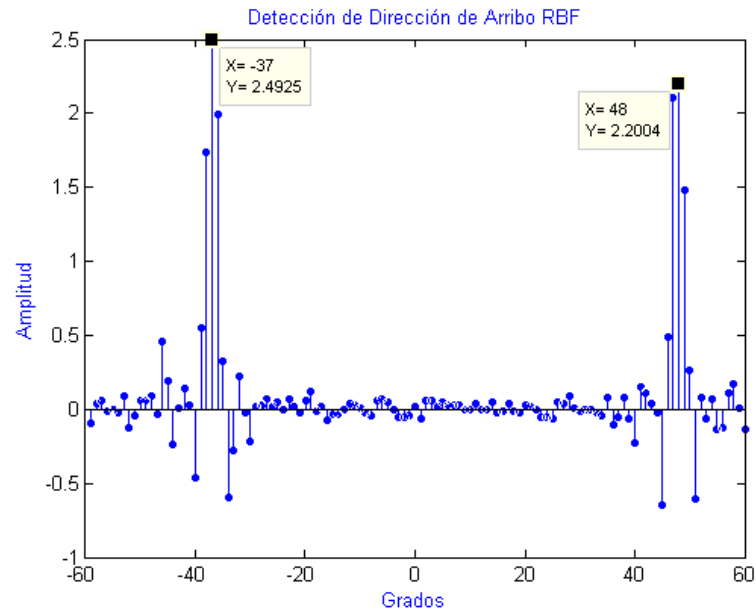


Figura 4.10: RBF $\theta_1 = -37$ y $\theta_2 = 48$.

Multiple Signal Classification

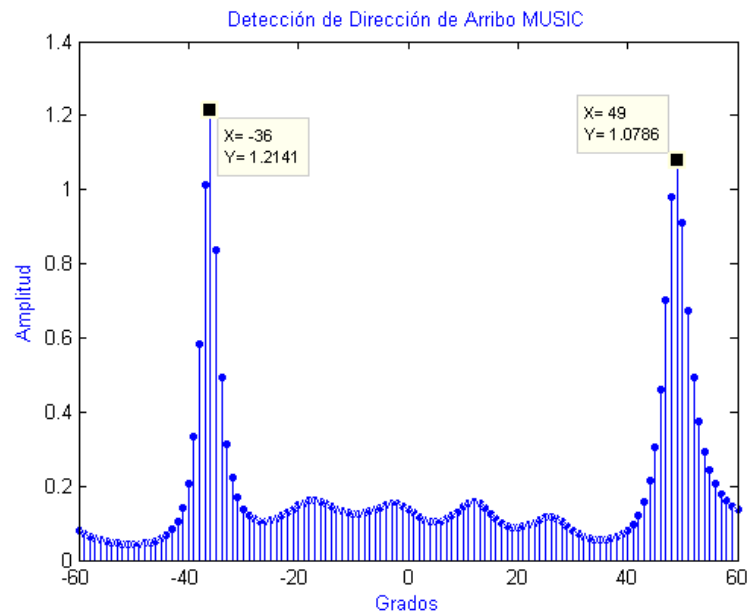


Figura 4.11: MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.

Deterministic Maximum Likelihood

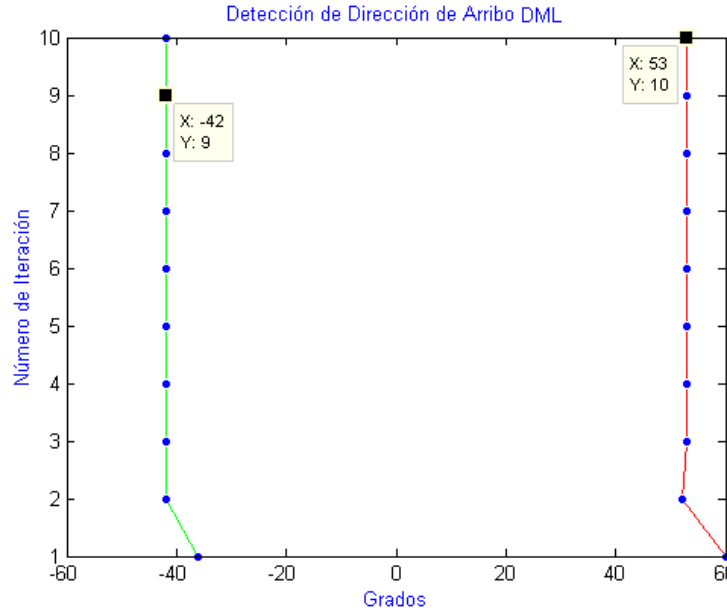


Figura 4.12: DML $\theta_1 = -37$ y $\theta_2 = 48$.

Evidentemente, por lo que se observa de las redes neuronales (4.9 y 4.10) se puede decir que las redes han sido bien entrenadas para la detección de dos ángulos de arribo, ya que compensa la simulación al detectar específicamente los ángulos deseados, a su vez se observa que en condiciones idénticas el resultado para MUSIC expuesto en la figura 4.11, existe un pequeño corrimiento de 1° a la derecha en los ángulos de detección respectivos, aun así se considera una detección no alejada de los ángulos de arribo solicitados, más sin embargo, para la detección por DML (4.12), no se detecta lo esperado arrojando valores de $\theta_1 = -42^\circ$ y $\theta_2 = 53^\circ$ en la última iteración, por ello se optó a generar otras condiciones para realizar conclusiones acertadas.

4.2.3. 2 señales Incidentes con separación intermedia

Ahora bien, se procede a generar las simulaciones para ver qué cambio ocurre si los ángulos de arribo están un poco más cercano uno con otro, específicamente

para los ángulos:

θ_1	θ_2
-7°	8°

Redes Perceptrón Multicapa

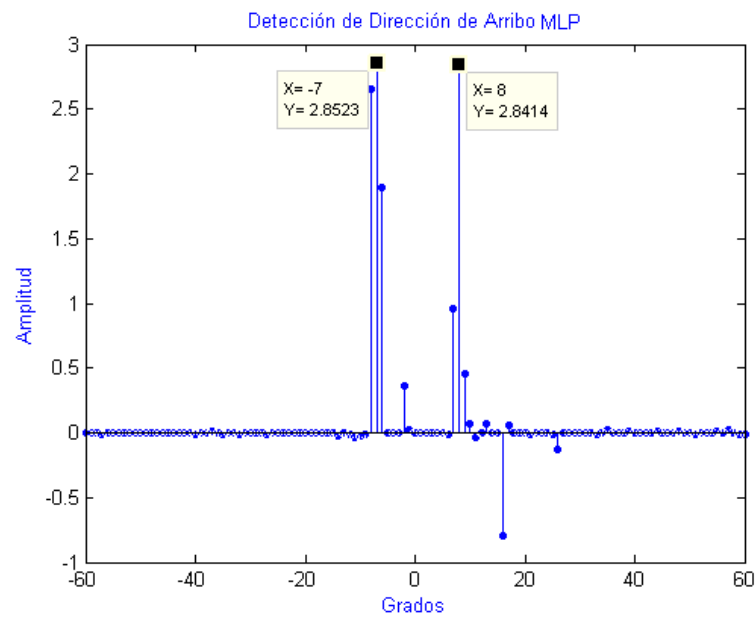


Figura 4.13: MLP $\theta_1 = -7$ y $\theta_2 = 8$.

Redes neuronales de base radial

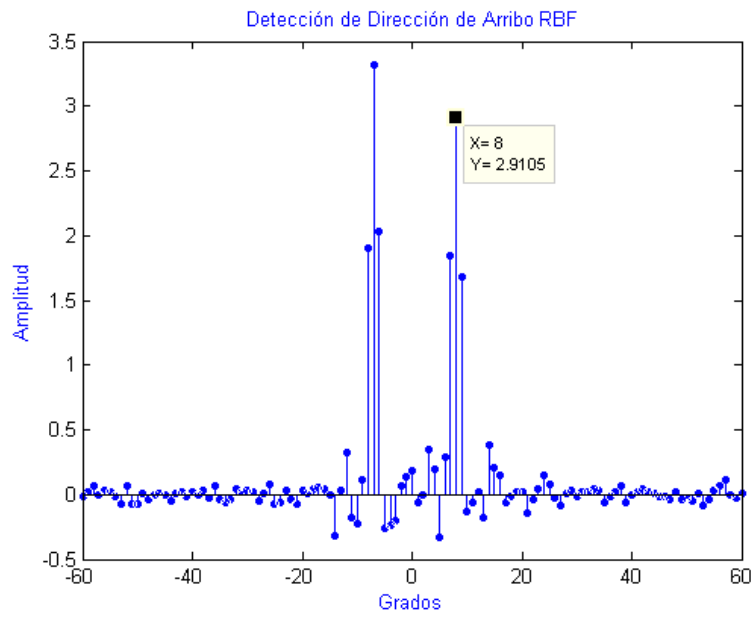


Figura 4.14: RBF $\theta_1 = -7$ y $\theta_2 = 8$.

Multiple Signal Classification

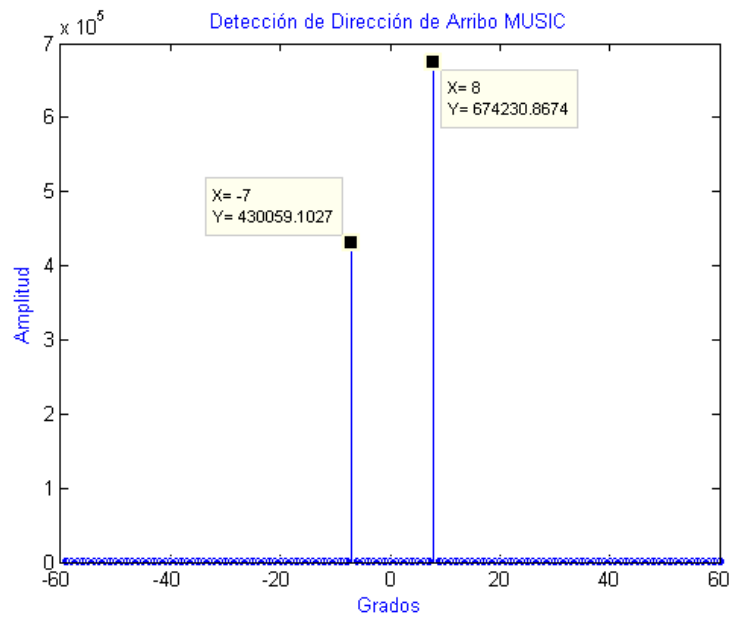


Figura 4.15: MUSIC $\theta_1 = -7$ y $\theta_2 = 8$.

Deterministic Maximum Likelihood

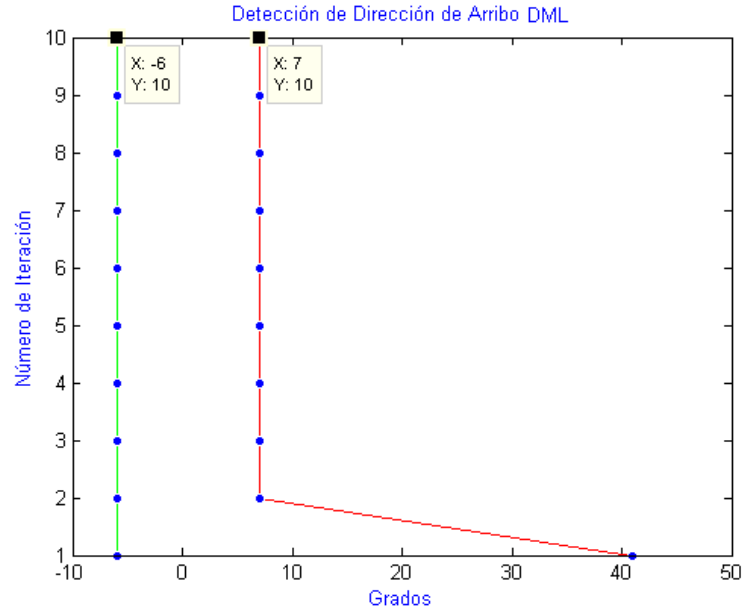


Figura 4.16: DML $\theta_1 = -7$ y $\theta_2 = 8$.

Podemos observar que tanto MLP (4.13), RBF (4.14) y MUSIC (4.15) se ajustan a los parámetros solicitados para la detección, estos otorgan un máximo en los θ_1 y θ_2 requeridos, a su vez se puede notar que DML (4.16) se acerca tímidamente a estos valores, con una variación de $\pm 1^\circ$ en la décima iteración, a pesar de ello, se adapta a los valores de ángulo de detección.

4.2.4. 2 señales Incidentes próximas

Por último, se decide hacer incidir dos señales que estén próximas entre sí con respecto al ángulo de llegada, para ver la máxima capacidad de detección, en cada simulación, para ello se establecieron las mismas condiciones óptimas, con ángulo de llegada por cada una de:

θ_1	θ_2
24°	25°

Redes Perceptrón Multicapa

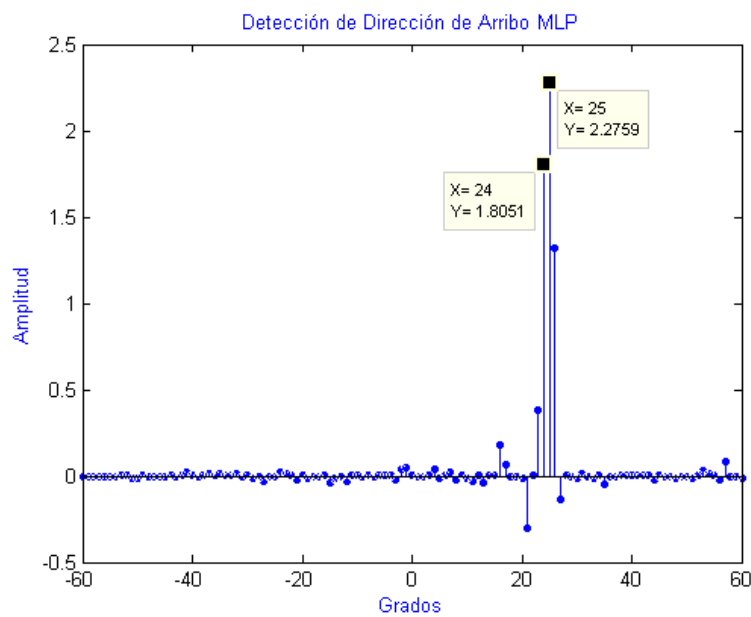


Figura 4.17: MLP $\theta_1 = 24$ y $\theta_2 = 25$.

Redes neuronales de base radial

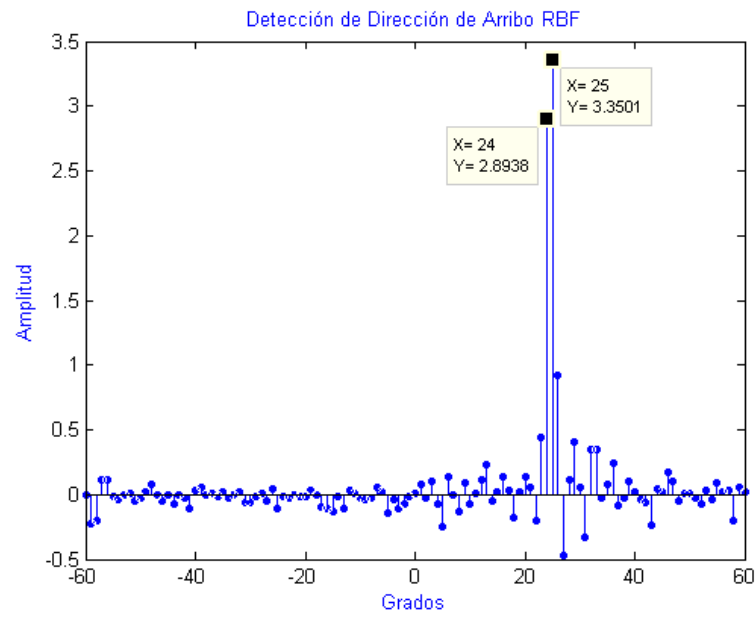


Figura 4.18: RBF $\theta_1 = 24$ y $\theta_2 = 25$.

Multiple Signal Classification

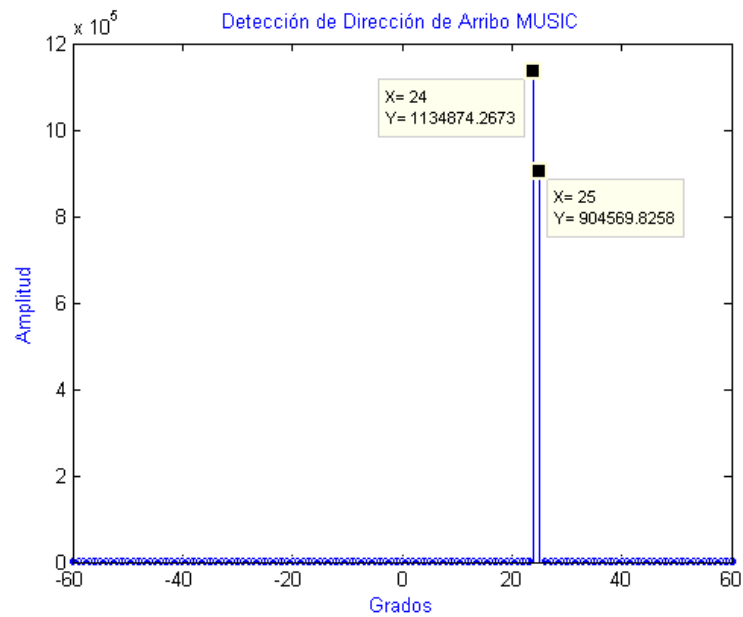


Figura 4.19: MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.

Deterministic Maximum Likelihood

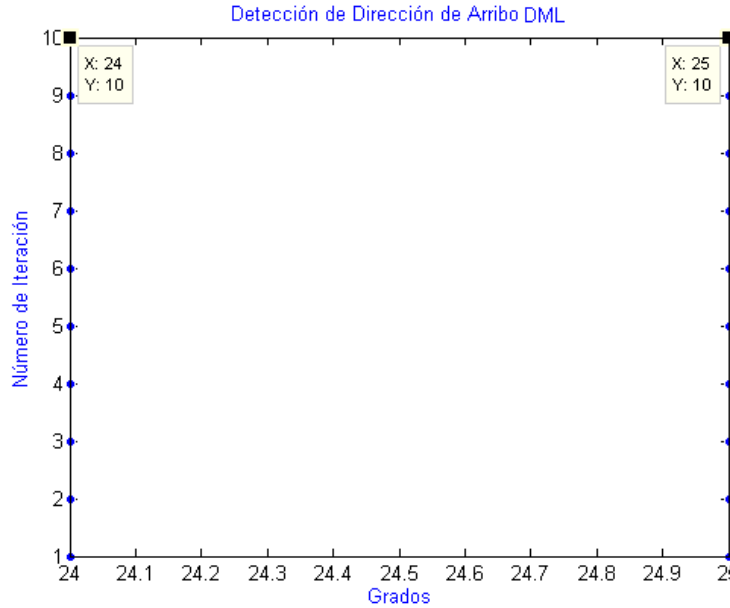


Figura 4.20: DML $\theta_1 = 24$ y $\theta_2 = 25$.

Es el mejor desempeño mostrado por cada una de las detecciones de arribo, donde DML(4.20) destaca ya que no tiene diferenciación de amplitud, para las demás gráfica de detecciones se observa como MLP(4.17) posee tres detecciones puntantes pero son $\theta_1 = 24$ y $\theta_2 = 25$ las que llevan ventaja con respecto a la detección siguiente, RBF (4.18) y MUSIC (4.19) están perfectamente adaptadas para sensor dos señales próximas entre sí.

4.2.5. 4 señales Incidentes

Ahora se incide 4 señales que están prácticamente equidistantes, y en condiciones ideales, para evaluar el comportamiento de las redes neuronales y de los algoritmos, así se decide quien de estas tiene mayor eficiencia.

θ_1	θ_2	θ_3	θ_4
-58°	-24°	17°	58°

Redes Perceptrón Multicapa

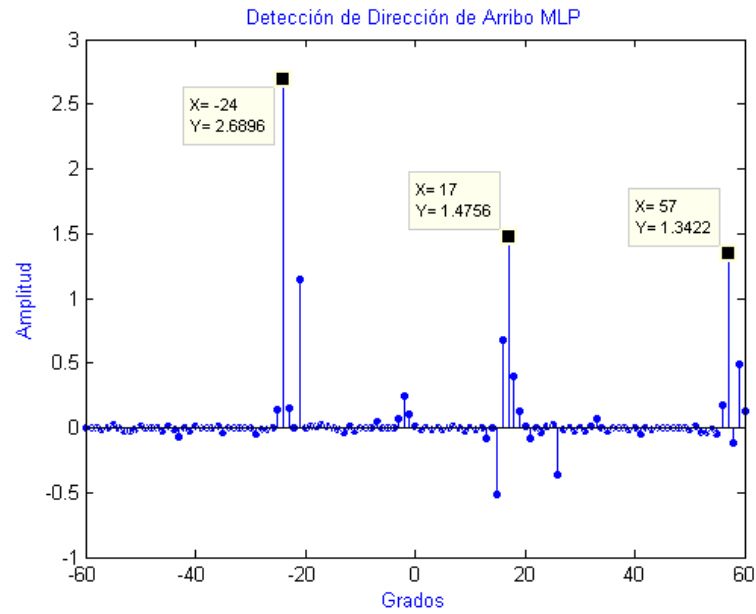


Figura 4.21: MLP $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Redes neuronales de base radial

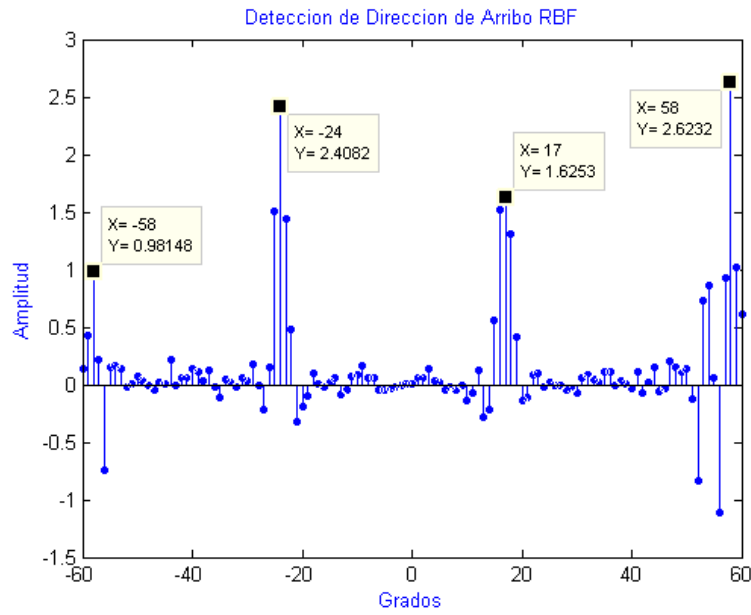


Figura 4.22: RBF $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Multiple Signal Classification

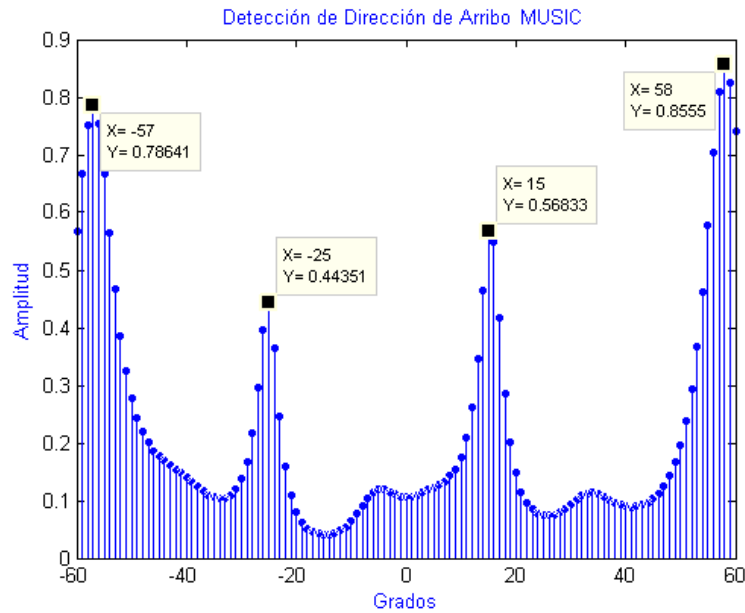


Figura 4.23: MUSIC $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Deterministic Maximum Likelihood

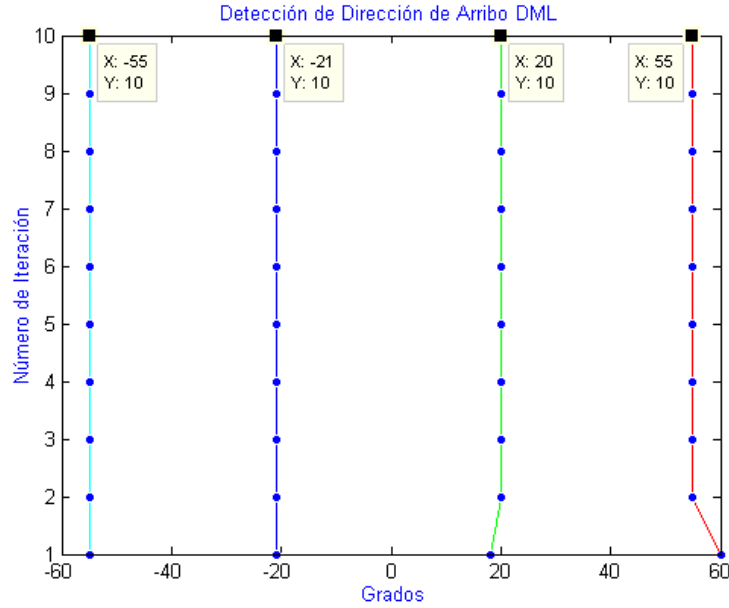


Figura 4.24: DML $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Podemos observar como MLP (4.21) detecta solo tres de las 4 señales de arribo eliminando por completo la detección del ángulo $\theta_1 = -58$, se podría inferir que RBF(4.22) y MUSIC (4.23), detectan los ángulos deseados, a pesar que esta última, tiene pequeñas variaciones en cada θ_i , no así ocurre en las redes neuronales de base radial, aunque se nota un mayor alzamiento de los ángulos próximos al ángulo de arribo; por su parte DML (4.24), tiene una noción de los ángulos de llegada, ya que, tiene una pequeña tolerancia de aproximadamente $\pm 3^\circ$ o $\pm 4^\circ$, esto nos indica que en cierto modo DML tiene incertidumbre considerable para 4 señales.

4.2.6. 7 señales Incidentes

Una vez hecho todos estos posibles diseños de ambientes de señales se optó por ver una mayor complejidad en las detecciones, es por ello que se han introducido 7 señales incidentes con sus respectivos ángulos de incidencia, así observaremos

cómo se comportan tanto las redes neuronales como los algoritmos de detección, viendo el grado de desenvolvimiento ante esta situación.

θ_1	θ_{12}	θ_3	θ_4	θ_5	θ_6	θ_7
-57°	-53°	-32°	-16°	5°	22°	53°

Redes Perceptrón Multicapa

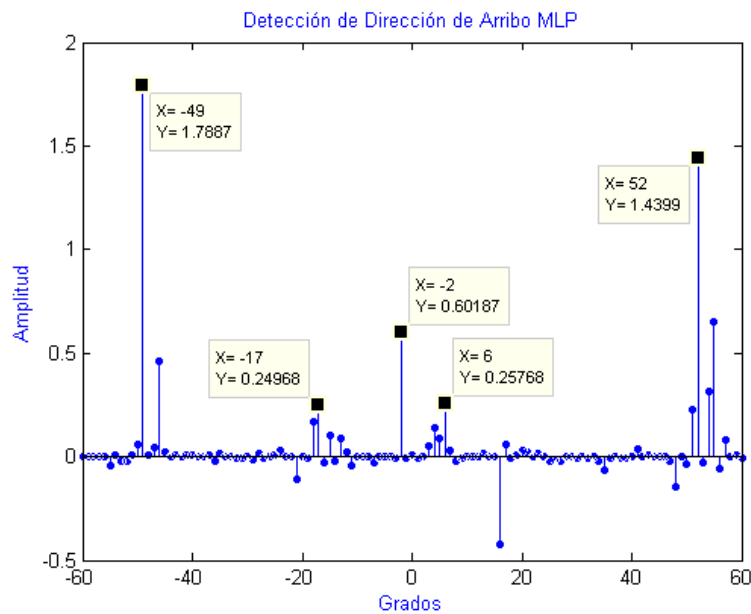


Figura 4.25: MLP $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Redes neuronales de base radial

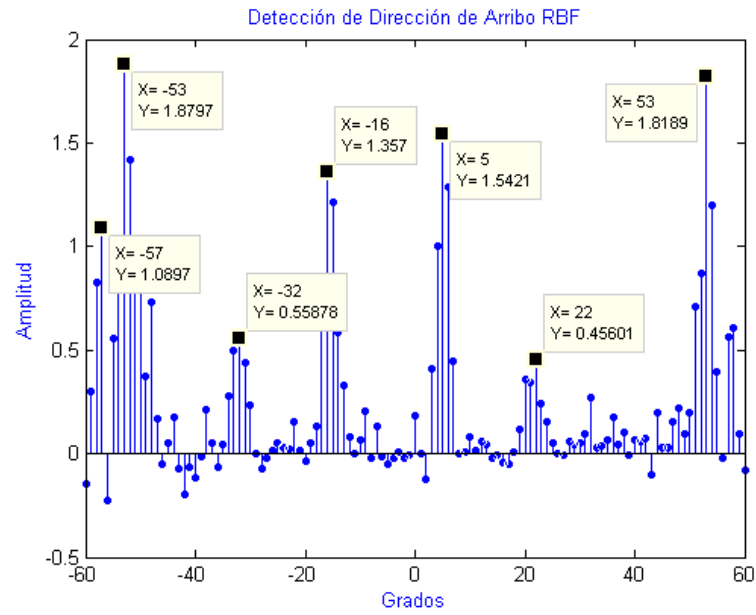


Figura 4.26: RBF $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Multiple Signal Classification

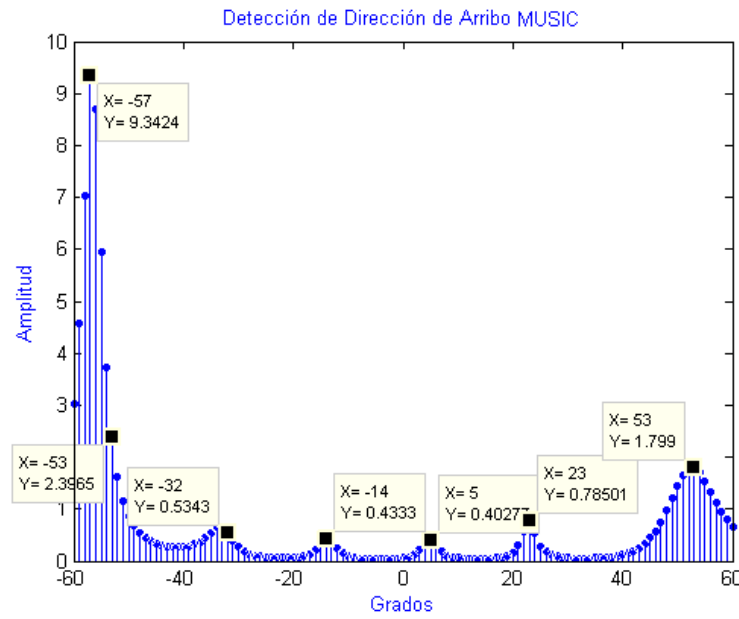


Figura 4.27: MUSIC $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Deterministic Maximum Likelihood

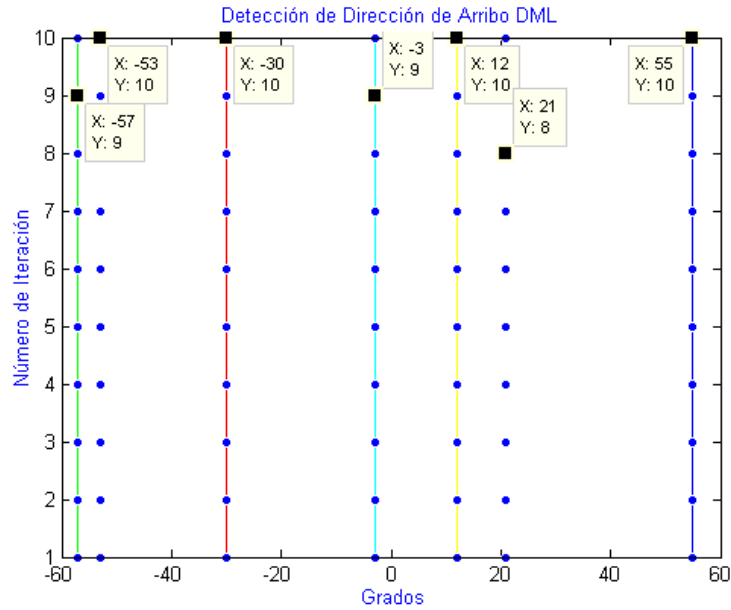


Figura 4.28: DML $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Sin duda la red Neuronal MLP (4.25) presenta un bajo rendimiento para 7 señales con ángulos de arribo separados, ya que no consigue detectar sino hasta 4 direcciones de arribo, y sin embargo, no son precisas además de adicionar una no existente como lo es para -2° , se recuerda que estamos en el caso idóneo, es decir que con presencia de ruido se espera que su rendimiento no sea aceptable. para las redes neuronales de base radial RBF (4.26) se puede observar como se ha detectado cada uno de los ángulos, adicional a estos, se genera levantamientos de detección cercanos a θ_i , pero manteniéndose por debajo del nivel detectado, en MUSIC (4.27) se observan los picos de detección casi exactos, solo que por la cercanía entre $\theta_1 = -57$ y $\theta_2 = -53$ estos sufren una unión en la detección, por ello se podría decir que se suprime, , además es importante denotar que solo un pico despunta $\theta_1 = -57$ la tolerancia angular de arribo, es de aproximadamente un grado, por lo que se puede decir que no existe una incertidumbre perjudicial. En DML(4.28) se percata que detecta 7 ángulos, de los cuales 5 están relativamente cerca admitiendo una tolerancia de $\pm 2^\circ$.

4.3. Resultados con relación señal a ruido de 20 dB

Una vez analizados los casos en óptimas condiciones, ahora se hará lo propio con relación señal a ruido de 20 dB de esta manera veremos cómo afecta el ruido en las señales incidentes.

4.3.1. 1 señal Incidente

θ	
-15°	46°

■ Redes Perceptrón Multicapa

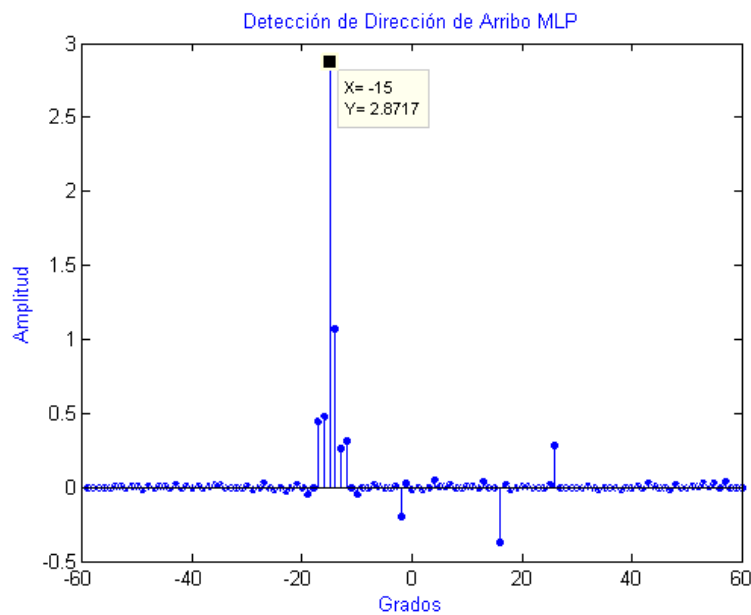
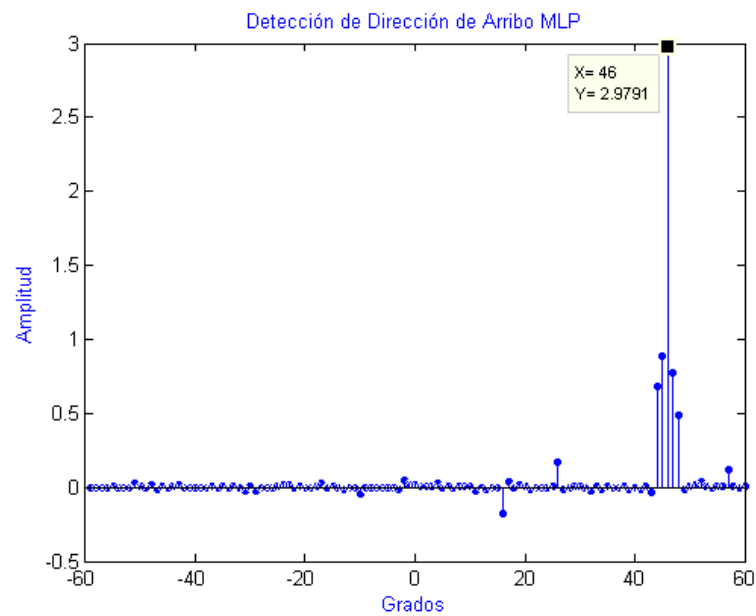
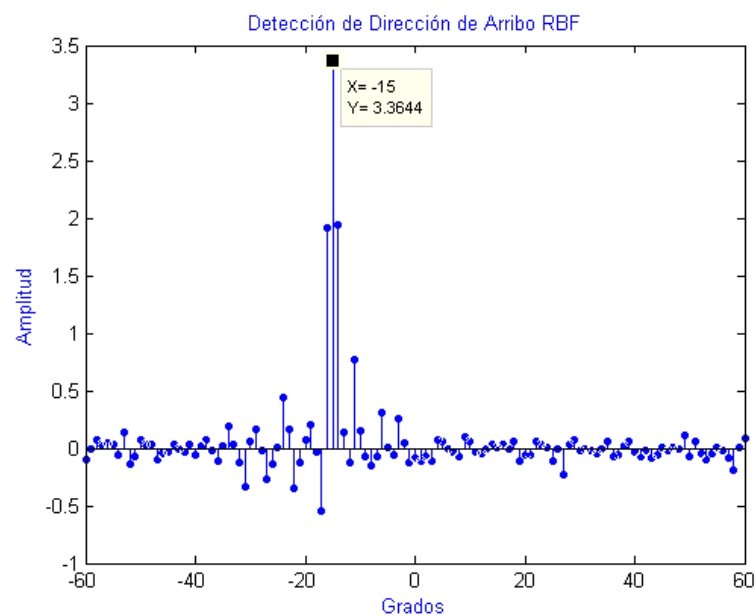


Figura 4.29: MLP $\theta = -15$.

Figura 4.30: MLP $\theta = 46$.

■ Redes Neuronales de Base Radial

Figura 4.31: RBF $\theta = -15$.

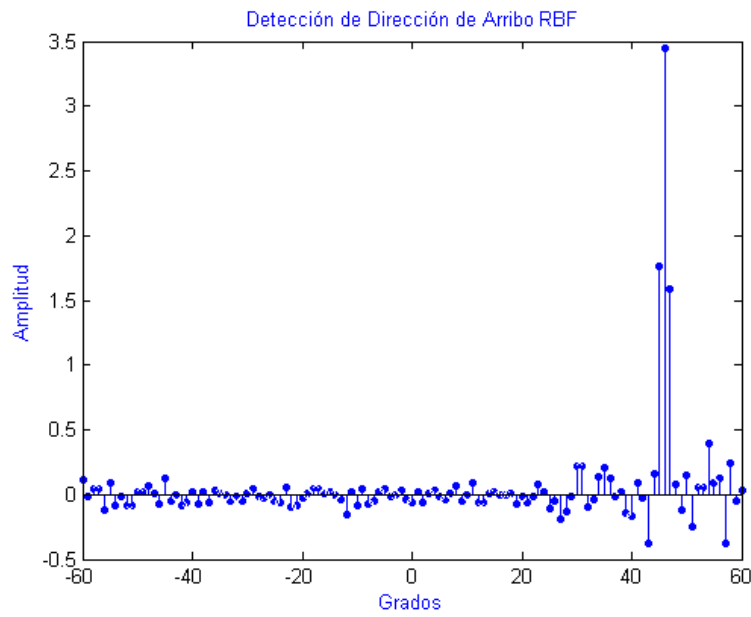
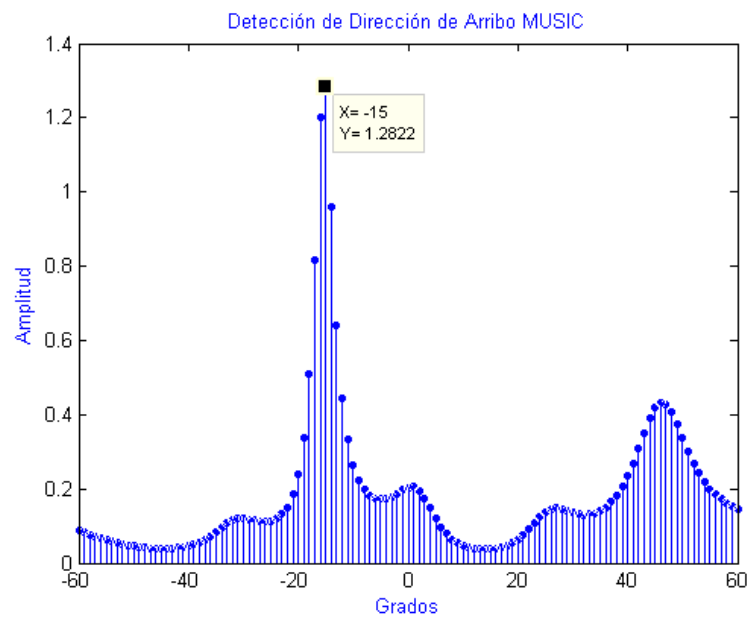
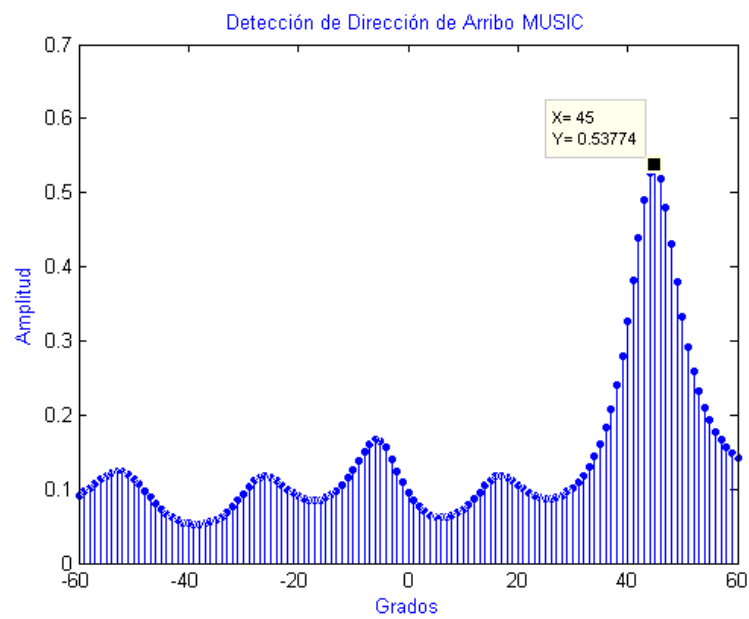


Figura 4.32: RBF $\theta = 46$.

- Multiple Signal Classification

**Figura 4.33:** MUSIC $\theta = -15$.**Figura 4.34:** MUSIC $\theta = 46$.

■ Deterministic Maximum Likelihood

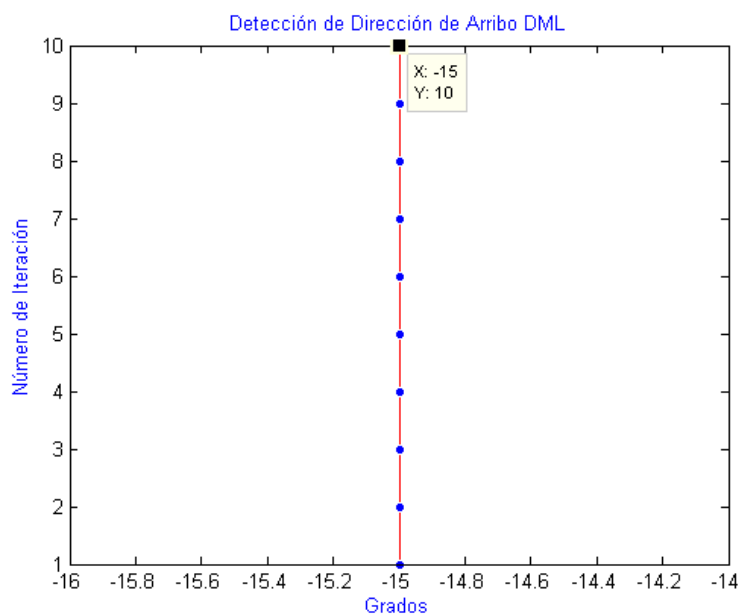


Figura 4.35: DML $\theta = -15$.

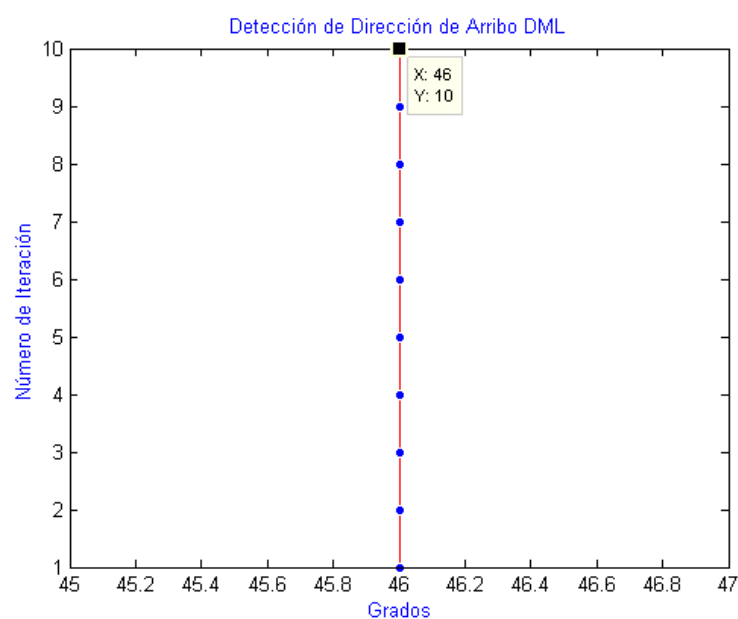


Figura 4.36: DML $\theta = 46$.

Podemos ver como las redes neuronales MLP (4.29 y 4.30) y RBF (4.31 4.32) tienen las mismas características que cuando la señal no posee ruido, en MUSIC podemos observar que cuando la señal tiene un solo ángulo de arribo se observa picos mínimos que bien pueden ser detectados pero tienen gran diferencia con respecto al ángulo detectado como se observa en las figuras (4.33) y (4.34), ahora en DML (4.35 y 4.36) si se muestra los ángulos de arribo exactos, sin incertidumbre en estos a pesar de tener ruido.

4.3.2. 2 señales Incidentes separadas

θ_1	θ_2
-37°	48°

Redes Perceptrón Multicapa

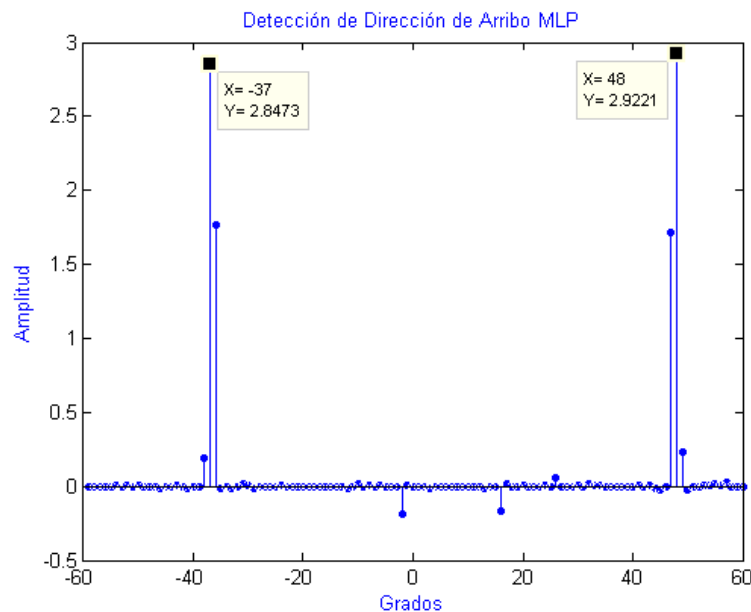


Figura 4.37: MLP $\theta_1 = -37$ y $\theta_2 = 48$.

Redes neuronales de base radial

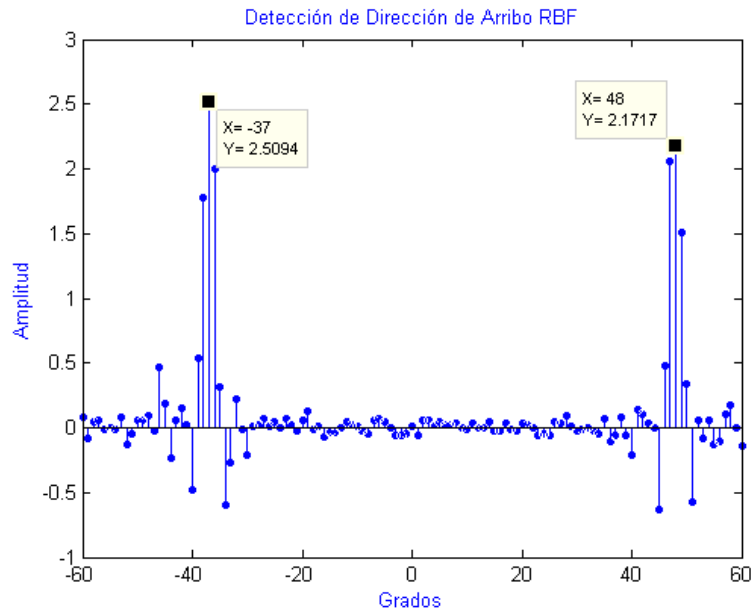


Figura 4.38: RBF $\theta_1 = -37$ y $\theta_2 = 48$.

Multiple Signal Classification

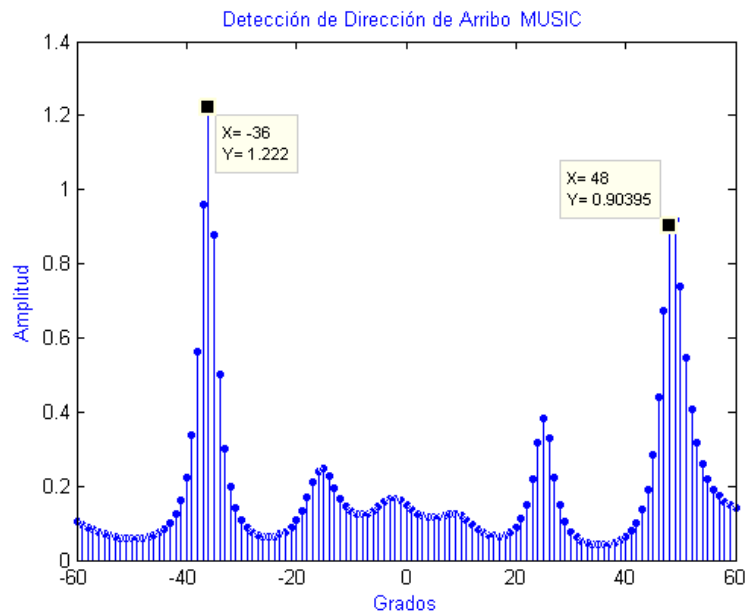


Figura 4.39: MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.

Deterministic Maximum Likelihood

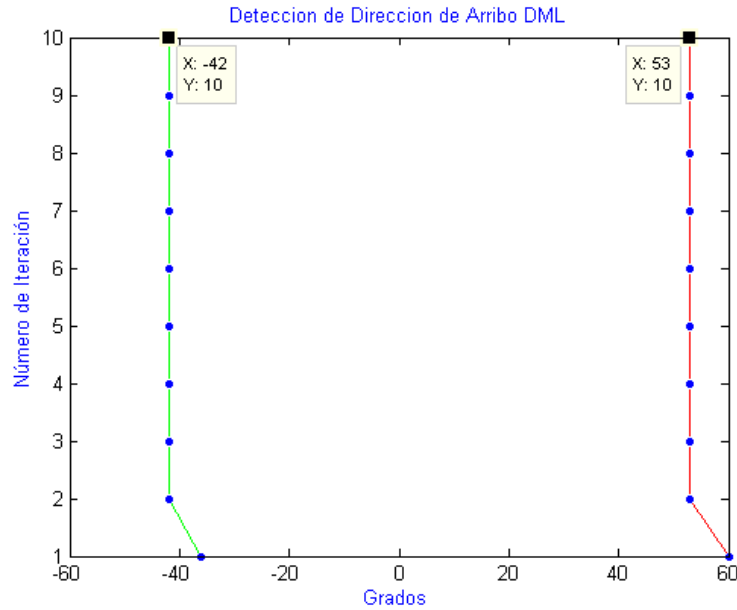


Figura 4.40: DML $\theta_1 = -37$ y $\theta_2 = 48$.

Notamos que las redes neuronales se comportan sin controversias detectando los ángulos de arribo expuestos en la simulación, así mismo ocurre para MUSIC (4.39) a pesar de tener un ángulo de arribo no detectado tan preciso como lo es $\theta_1 = -37$, que arroja $\theta = -36$, pero esto, no afecta la detección ya que se puede considerar un ángulo detectado, por su parte DML(4.40) no detecta los ángulos de arribo y sigue comportándose como en el caso ideal, por lo que se observa como se le es complicado detectar dos señales muy separadas.

4.3.3. 2 señales Incidentes con separacion intermedia

θ_1	θ_2
-7°	8°

Redes Perceptrón Multicapa

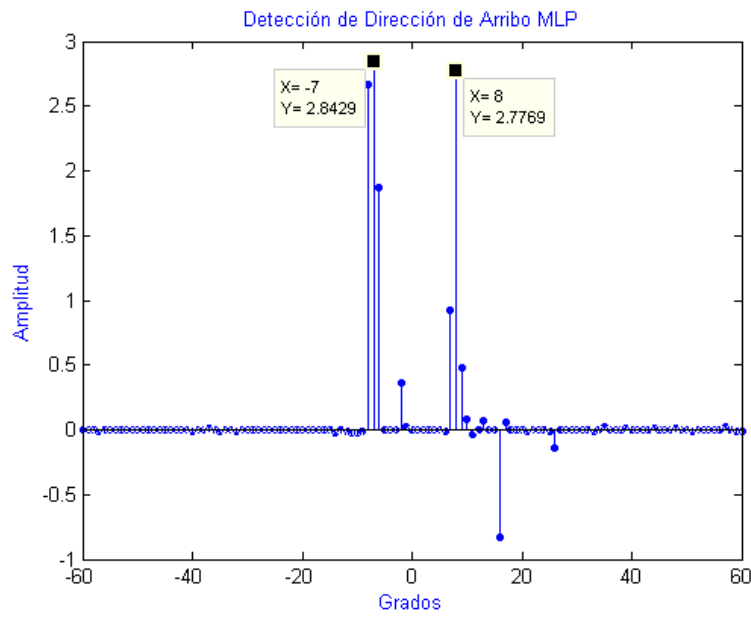


Figura 4.41: MLP $\theta_1 = -7$ y $\theta_2 = 8$.

Redes neuronales de base radial

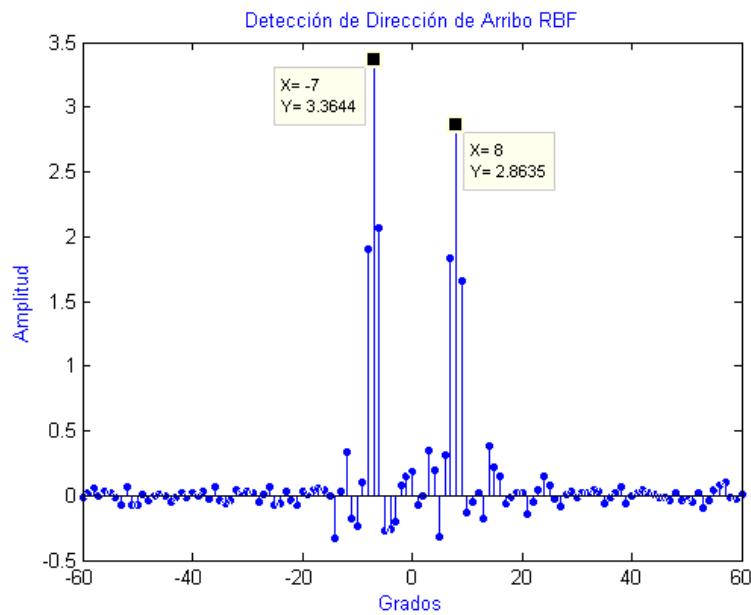


Figura 4.42: RBF $\theta_1 = -7$ y $\theta_2 = 8$.

Multiple Signal Classification

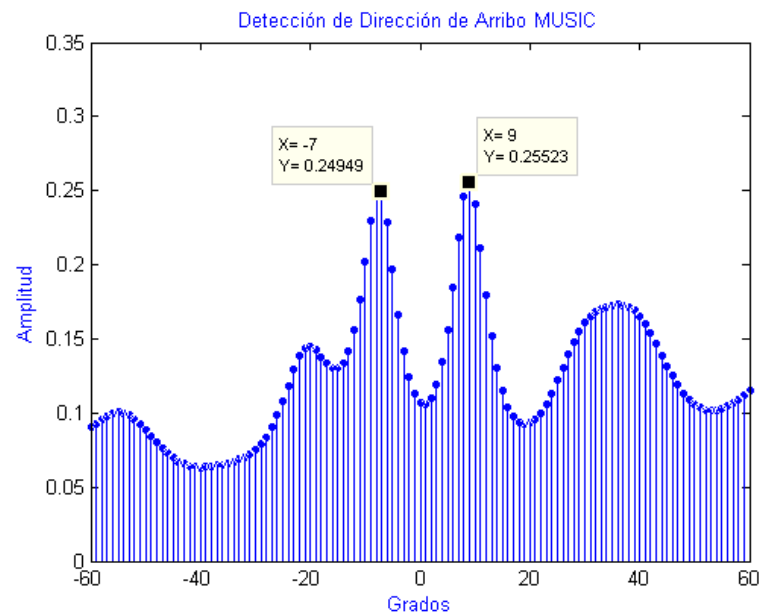


Figura 4.43: RBF $\theta_1 = -7$ y $\theta_2 = 8$.

Deterministic Maximum Likelihood

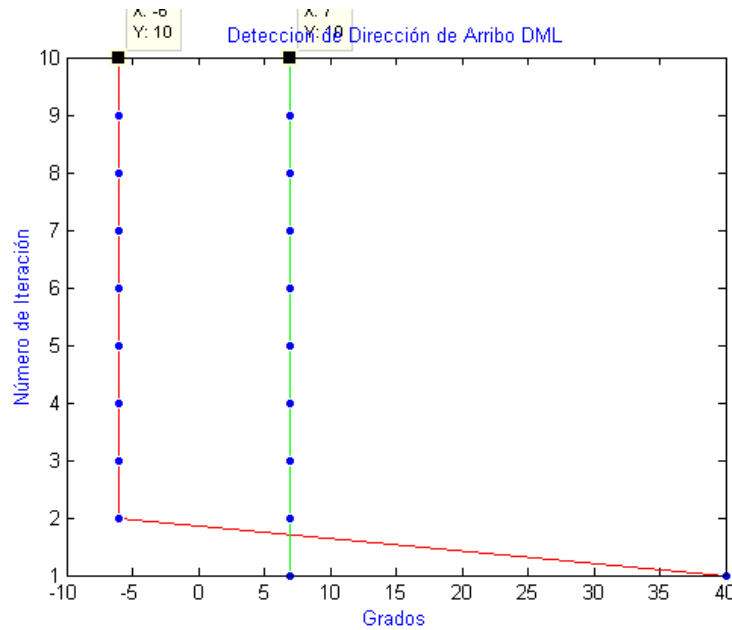


Figura 4.44: DML $\theta_1 = -7$ y $\theta_2 = 8$.

Se detalla como las redes neuronales MLP (4.41) y RBF (4.42) cumplen con la detección del ángulo de arribo deseado, sabiendo que el primero está despuntando con respecto a los ángulos próximos específicamente en $\theta_1 = -7$ donde se observa un levantamiento inferior para los ángulos -8° y -6° , MUSIC (4.43) por su parte detecta los ángulos de arribo uno exacto y el otro con una incertidumbre de 1° , en DML (4.44) vemos que se comporta igual que en caso ideal, dando valores de $\theta \pm 1^\circ$ en la décima iteración.

4.3.4. 2 señales Incidentes próximas

θ_1	θ_2
24°	25°

Redes Perceptrón Multicapa

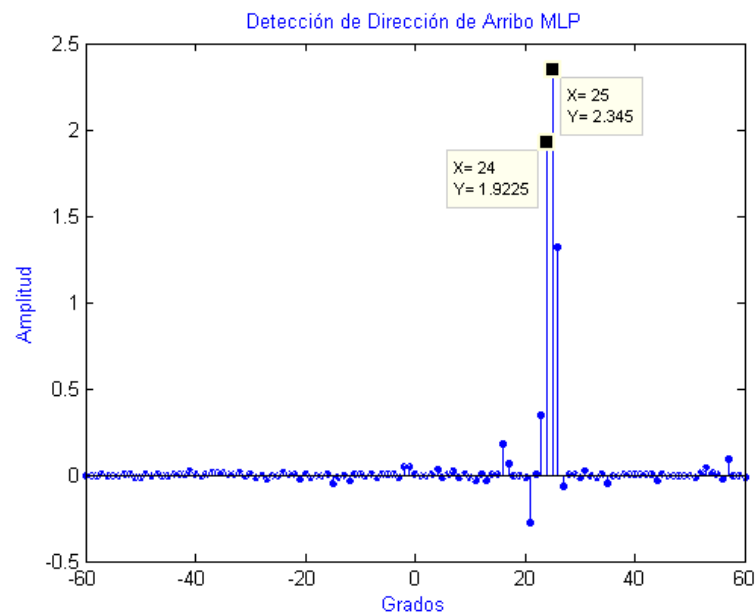


Figura 4.45: MLP $\theta_1 = 24$ y $\theta_2 = 25$.

Redes neuronales de base radial

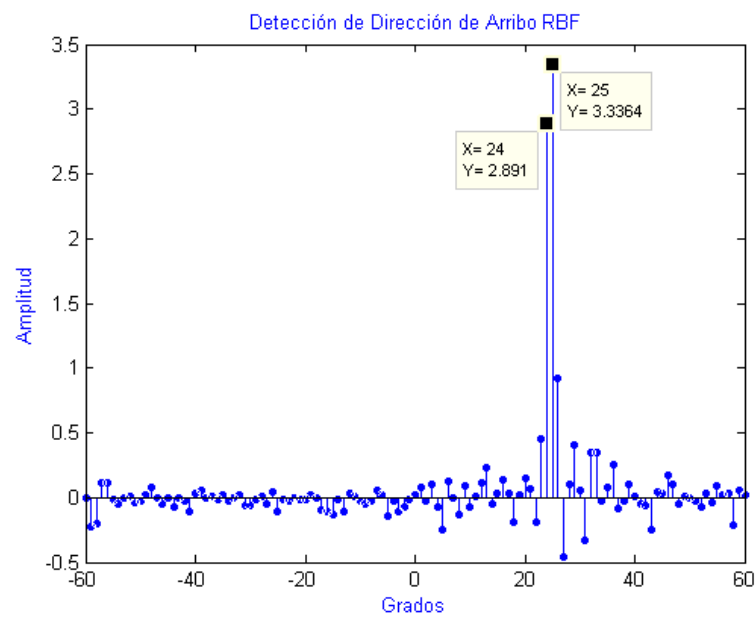


Figura 4.46: RBF $\theta_1 = 24$ y $\theta_2 = 25$.

Multiple Signal Classification

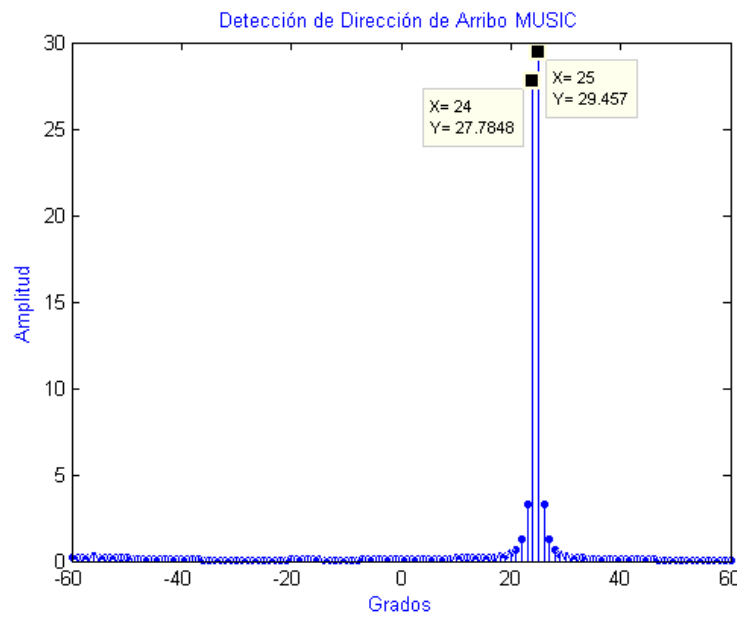


Figura 4.47: MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.

Deterministic Maximum Likelihood

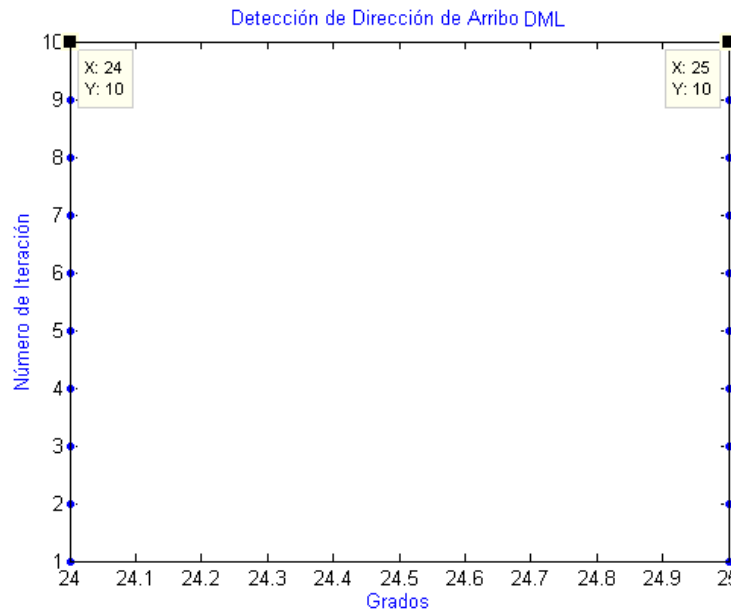


Figura 4.48: DML $\theta_1 = 24$ y $\theta_2 = 25$.

Se infiere que cuando las señales están sumamente próximas con respecto al ángulo de llegada, en un ámbito real con relación señal a ruido de 20 dB tanto las redes neuronales como los algoritmos para detección de ángulos de arribo se comportan de manera idéntica que en el caso idóneo, así lo observamos en las figuras arrojadas por MLP(4.45), RBF(4.46), MUSIC(4.47) y DML (4.48) donde cada simulación arroja los valores esperados.

4.3.5. 4 señales Incidentes

θ_1	θ_2	θ_3	θ_4
-58°	-24°	17°	58°

Redes Perceptrón Multicapa

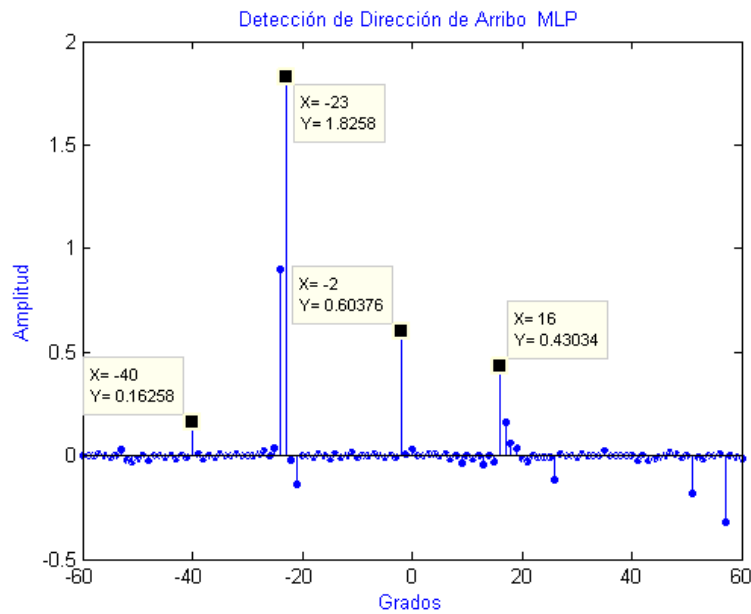


Figura 4.49: MLP $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Redes neuronales de base radial

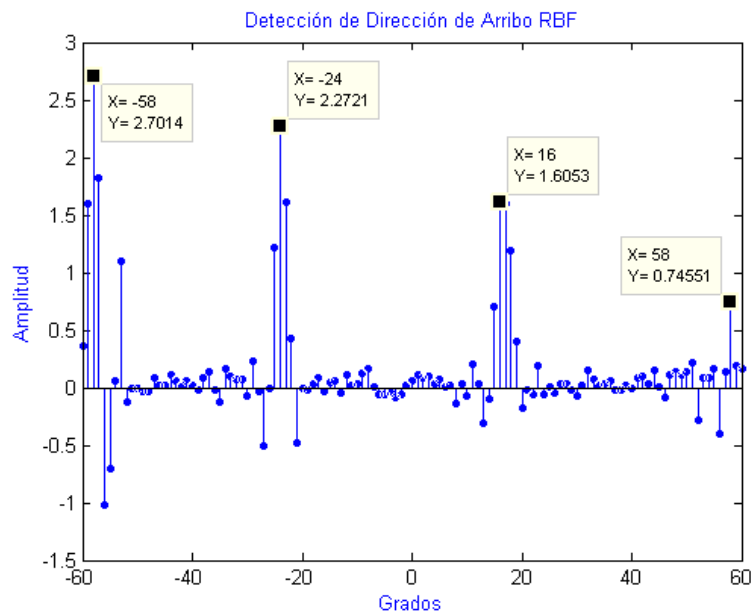


Figura 4.50: RBF $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Multiple Signal Classification

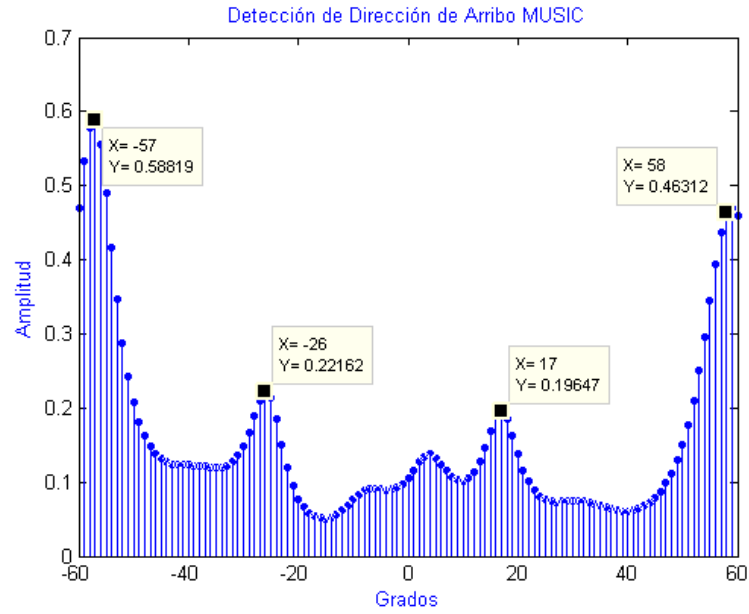


Figura 4.51: MUSIC $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Deterministic Maximum Likelihood

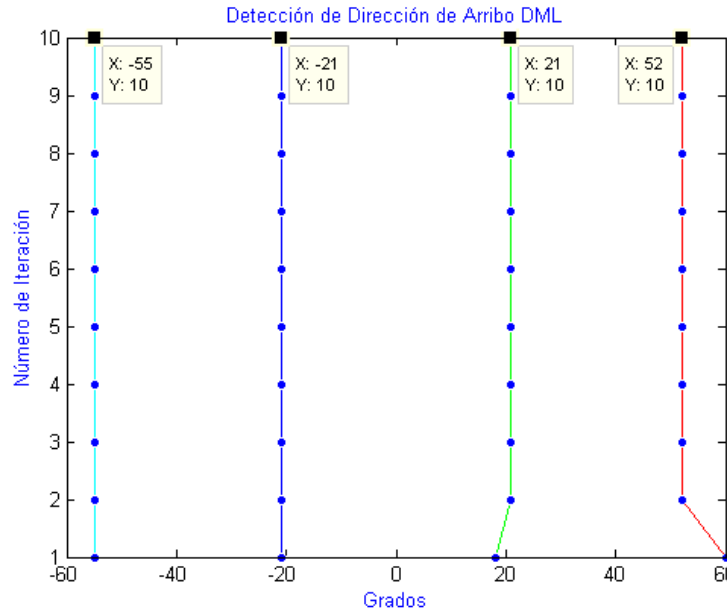


Figura 4.52: DML $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Las red neuronal MLP (4.49) no acierta sino dos ángulos de arribo con una incertidumbre de $\pm 1^\circ$, por otro lado la red neuronal RBF (4.50) da con los ángulos de llegada esperados, siendo $\theta_3 = 17$ el ángulo detectado asumiendo una tolerancia de $\pm 1^\circ$, mas sin embargo, se podría decir que es el que más rendimiento desempeña en esta ocasión junto con MUSIC (4.51) que detecta cuatro ángulos donde dos amplitudes despuntan con respecto a las otras dos, como lo es en $\theta_1 = -58$ y $\theta_4 = 58$, siendo el primero detectado para -57° como ocurre con $\theta_2 = -24$ donde se detecta es en -26° . DML (4.52) se comporta tal cual en el caso idóneo, donde los ángulos de arribo tienen una variación general de $\pm 6^\circ$ en el peor de los casos, pero con mucho mejor rendimiento que la red neuronal MLP.

4.3.6. 7 señales Incidentes

Redes Perceptrón Multicapa

θ_1	θ_{12}	θ_3	θ_4	θ_5	θ_6	θ_7
-57°	-53°	-32°	-16°	5°	22°	53°

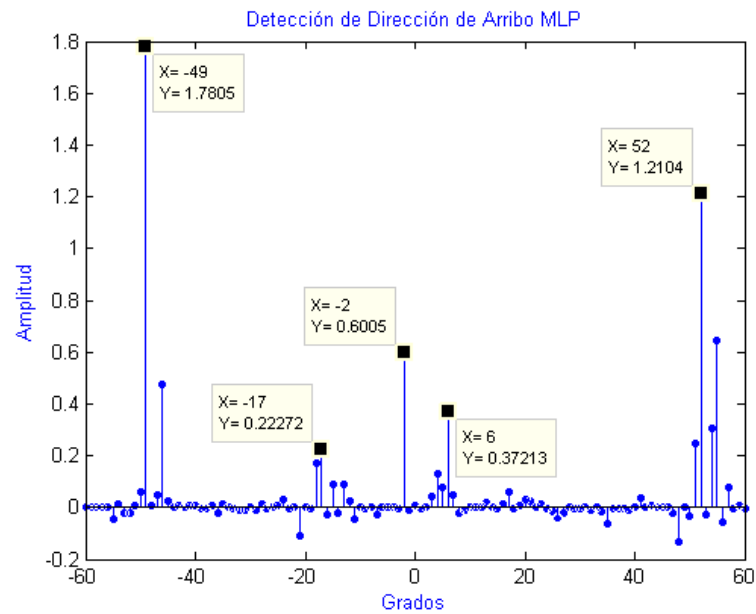


Figura 4.53: MLP $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Redes neuronales de base radial

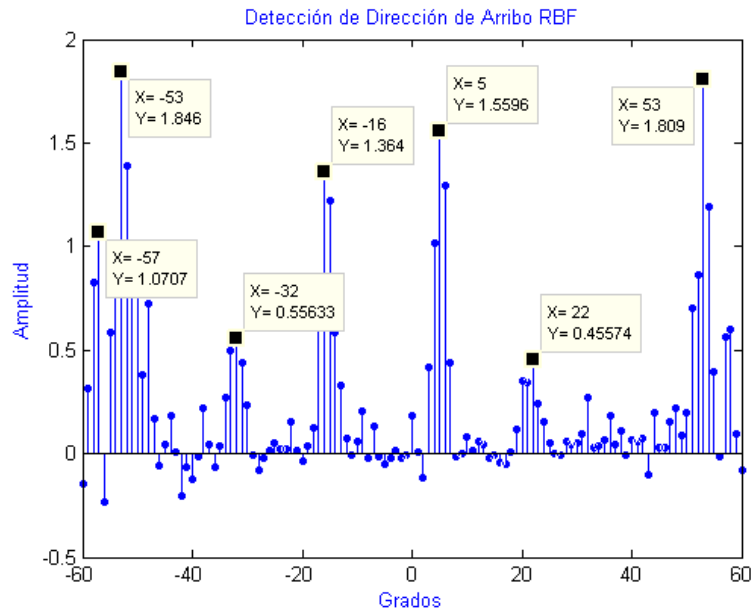


Figura 4.54: RBF $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Multiple Signal Classification

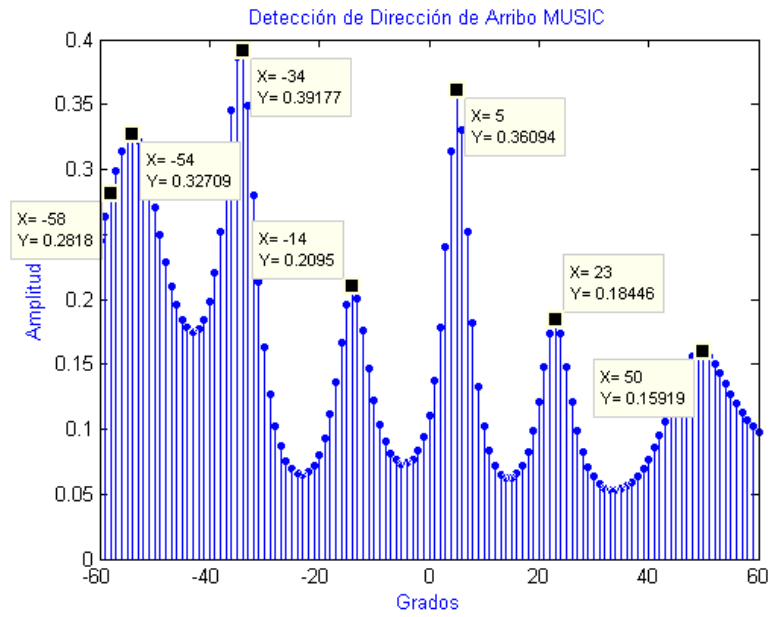


Figura 4.55: MUSIC $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Deterministic Maximum Likelihood

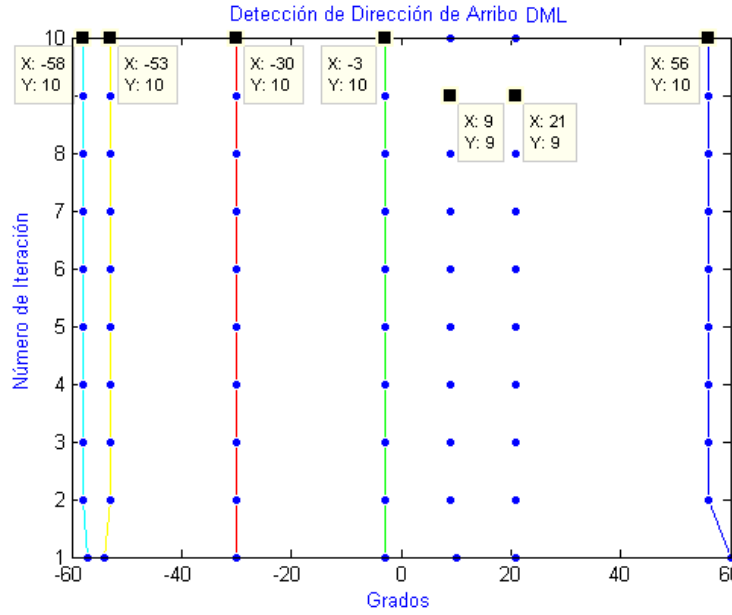


Figura 4.56: DML $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Vemos en MLP (4.53) que son los mismos resultados mostrados en el caso ideal, donde se dice que detecto cuatro señales aproximadamente parecidas al ángulo de llegada esperado, mientras que se sigue eliminando las mismas tres direcciones de arribo y se añade una detección no deseada en -2° nuevamente, a su vez en RBF (4.54) muestra las señales deseadas en los ángulos dados; MUSIC (4.55) detecta los mismos seis ángulos despuntando $\theta_1 = -57$ sobre los demás haciéndolo ver como si solo está detectando una sola señal, por otro lado DML (4.56) sigue teniendo los mismos desacierto en los ángulos $\theta_4 = -16^\circ$ y $\theta_5 = 5^\circ$ mostrando valores de -2° y 12° respectivamente.

4.4. Resultados con relacion señal a ruido de 5 dB

4.4.1. 1 señal Incidente

θ	
-15°	46°

Redes Perceptrón Multicapa

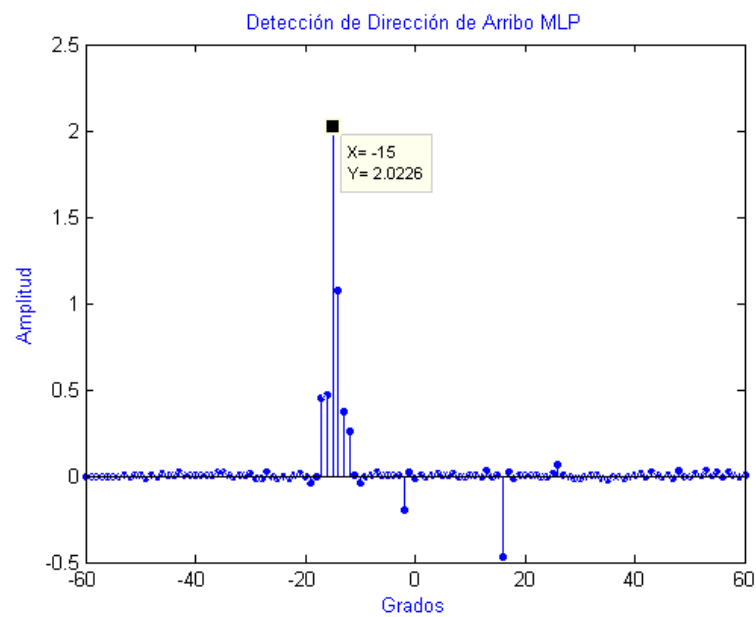


Figura 4.57: MLP $\theta = -15$.

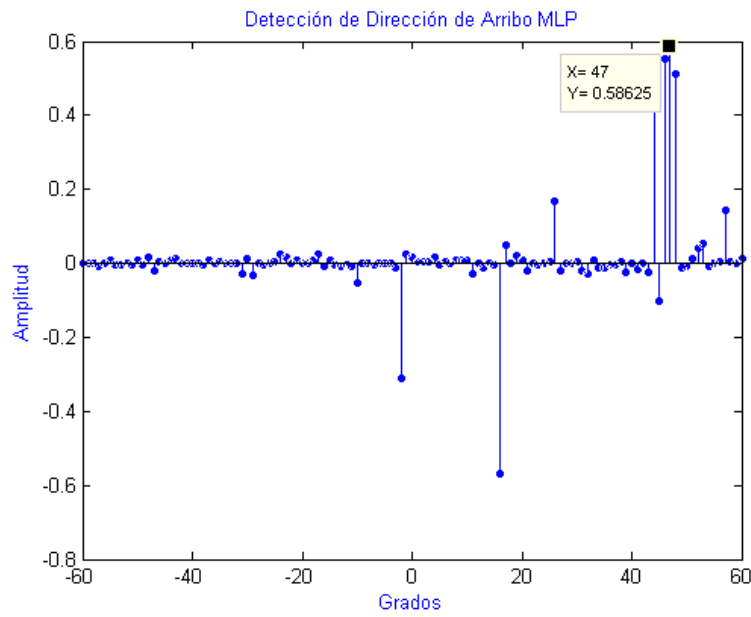


Figura 4.58: MLP $\theta = 46$.

Redes Neuronales de Base Radial

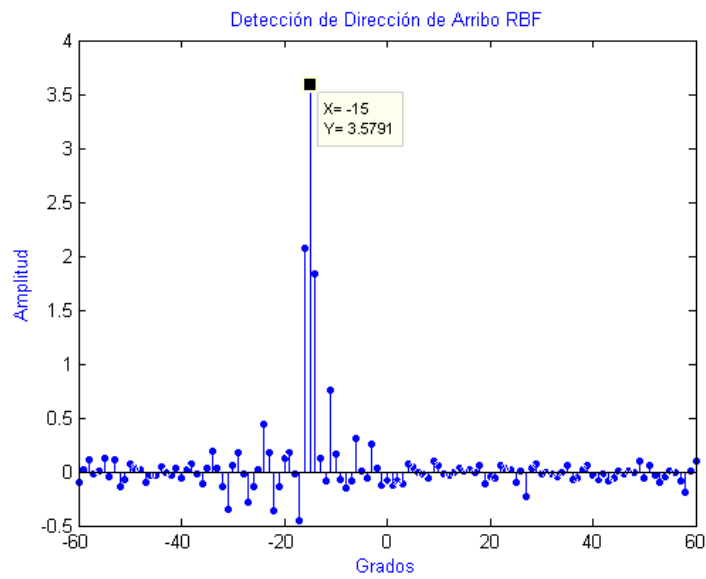
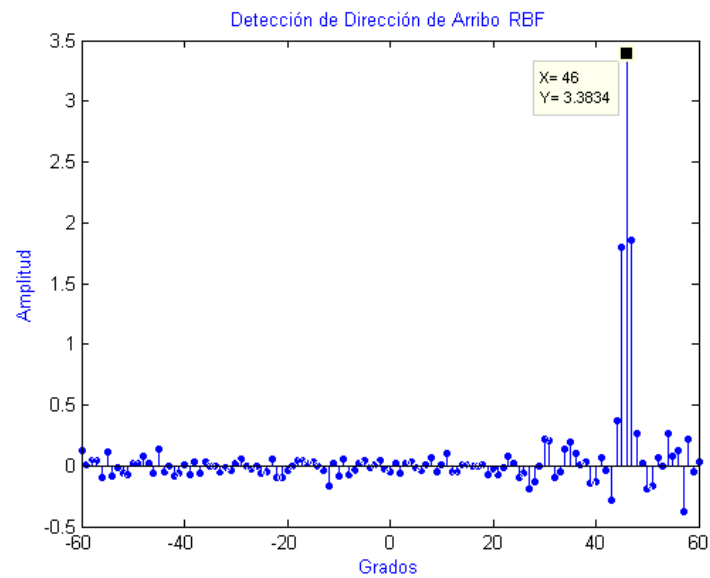
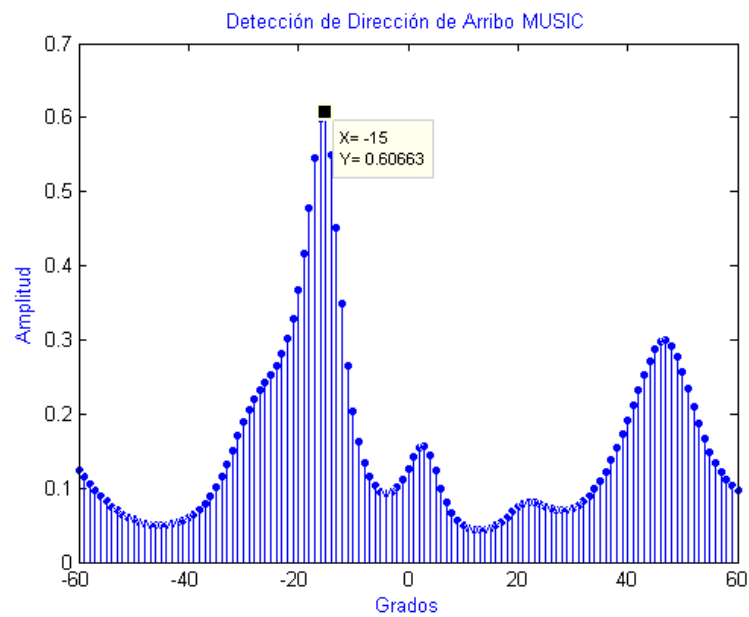


Figura 4.59: RBF $\theta = -15$.

**Figura 4.60:** RBF $\theta = 46$.

Multiple Signal Classification

**Figura 4.61:** MUSIC $\theta = -15$.

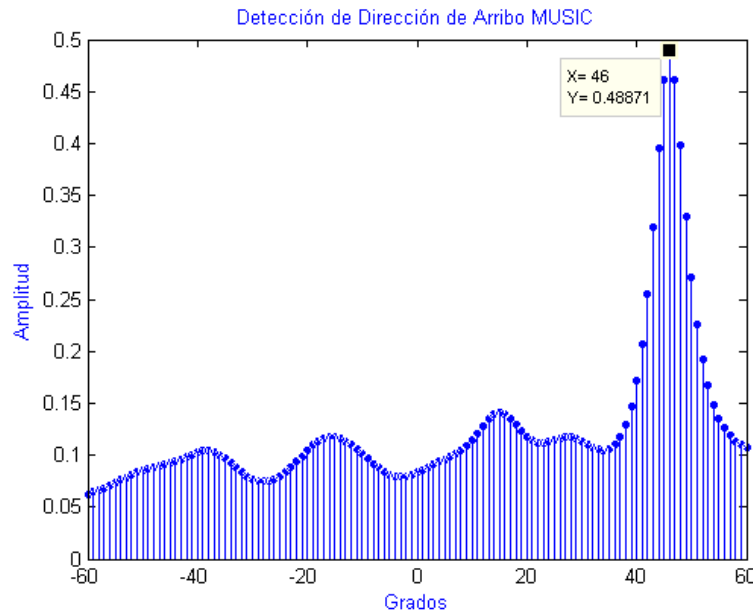


Figura 4.62: MUSIC $\theta = 46$.

Deterministic Maximum Likelihood

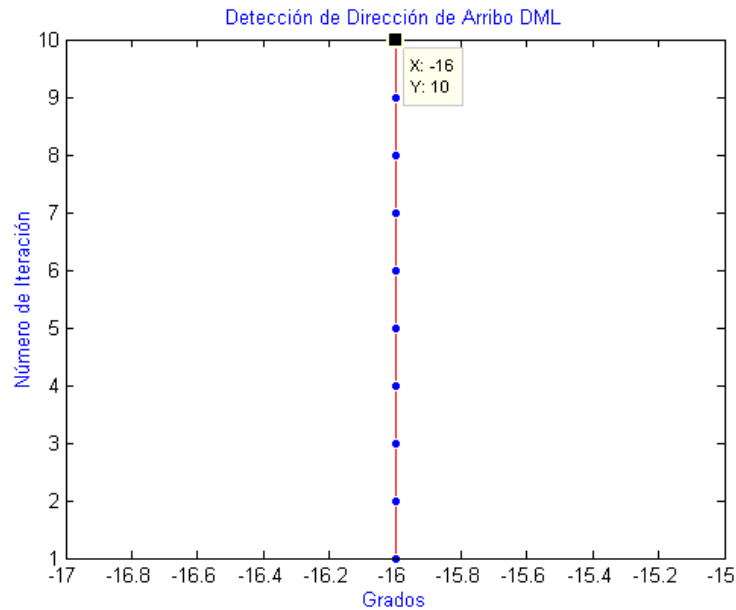
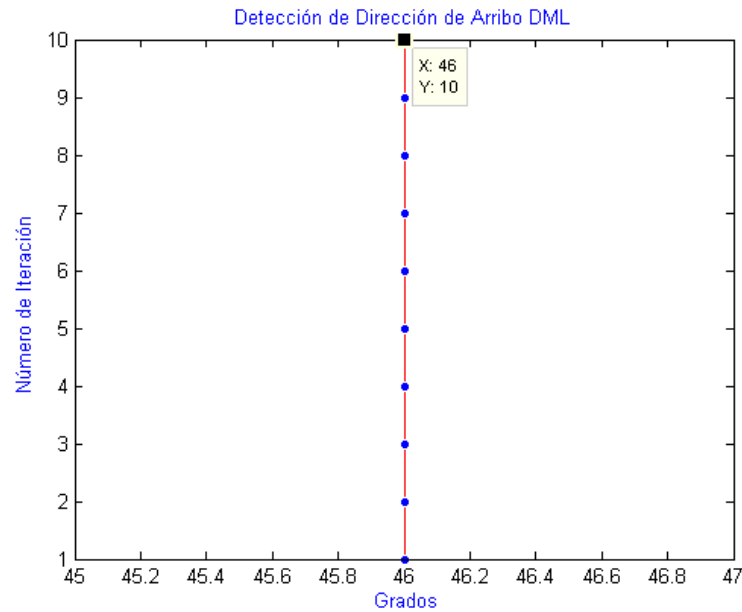


Figura 4.63: DML $\theta = -15$.

Figura 4.64: DML $\theta = 46$.

Se percibe que las redes neuronales RBF (4.59 y 4.60) está dispuestas a detectar una señal de arribo aun cuando el ruido en estas es grande y perjudicial, así mismo MLP (4.57 y 4.58) tiene un buen comportamiento para la detección de $\theta = -15^\circ$ y para $\theta = 46^\circ$ existen detecciones próximas que hacen que la amplitud mayor sea detectada en 47° ; MUSIC (4.61 y 4.62) hace las respectivas detecciones para cada ángulo de arribo por separado, así mismo se puede distinguir otros picos que podrían tomarse como una detección según los criterios que se fijen. DML(4.63 y 4.64) tiene una incertidumbre de $\pm 1^\circ$ en la detección de $\theta = -15^\circ$, aun así hace una buena detección para los ángulos de arribo estudiados en este caso con una señal de arribo.

4.4.2. 2 señales Incidentes separadas

θ_1	θ_2
-37°	48°

Redes Perceptrón Multicapa

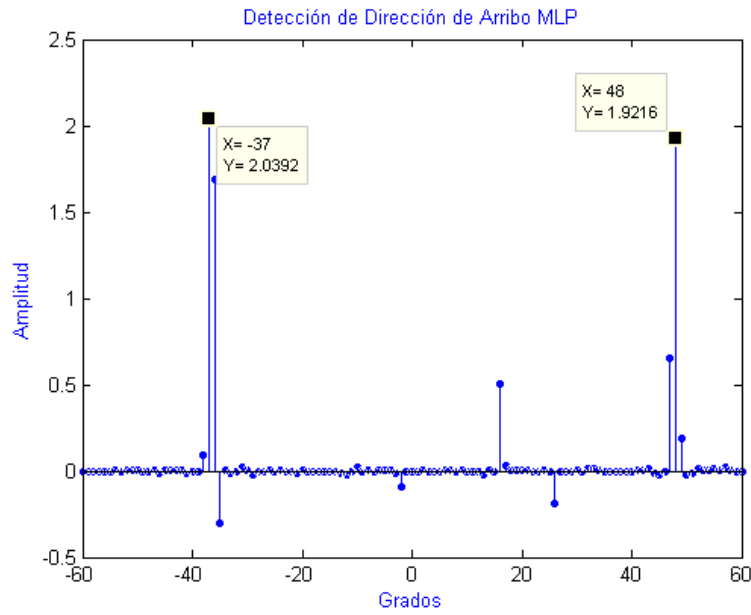


Figura 4.65: MLP $\theta_1 = -37$ y $\theta_2 = 48$.

Redes neuronales de base radial

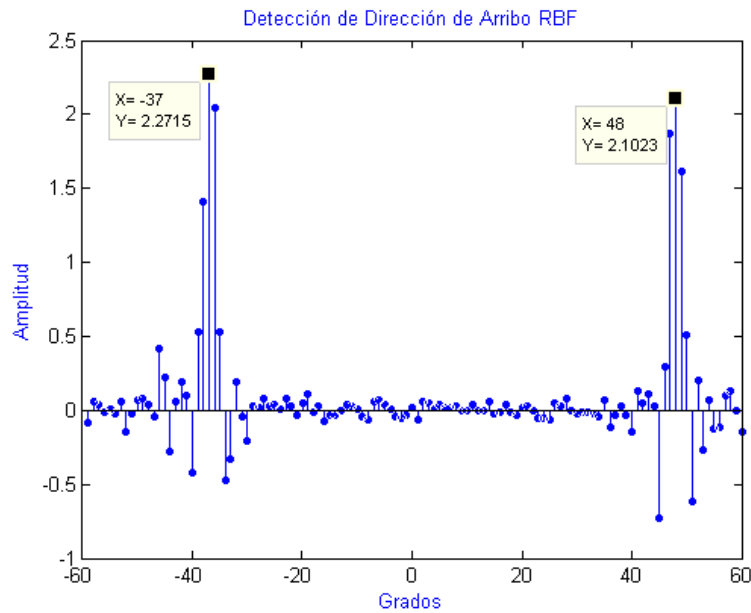


Figura 4.66: RBF $\theta_1 = -37$ y $\theta_2 = 48$.

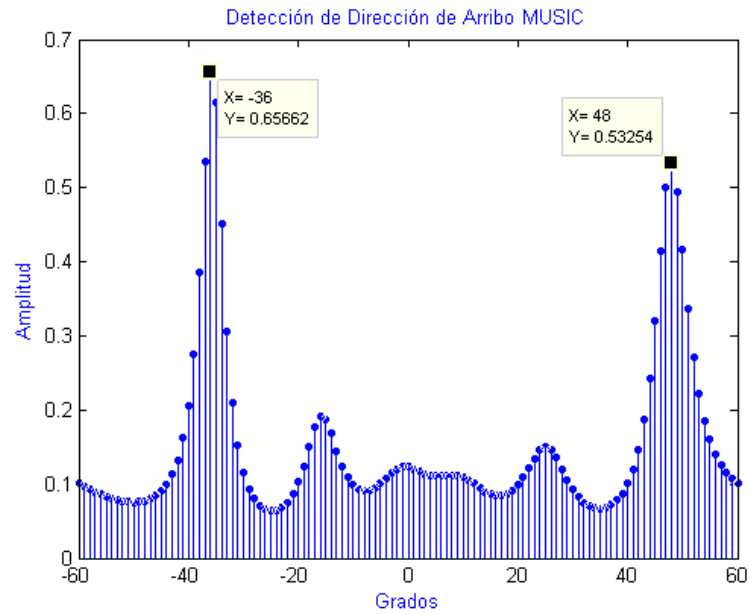
Multiple Signal Classification

Figura 4.67: MUSIC $\theta_1 = -37$ y $\theta_2 = 48$.

Deterministic Maximum Likelihood

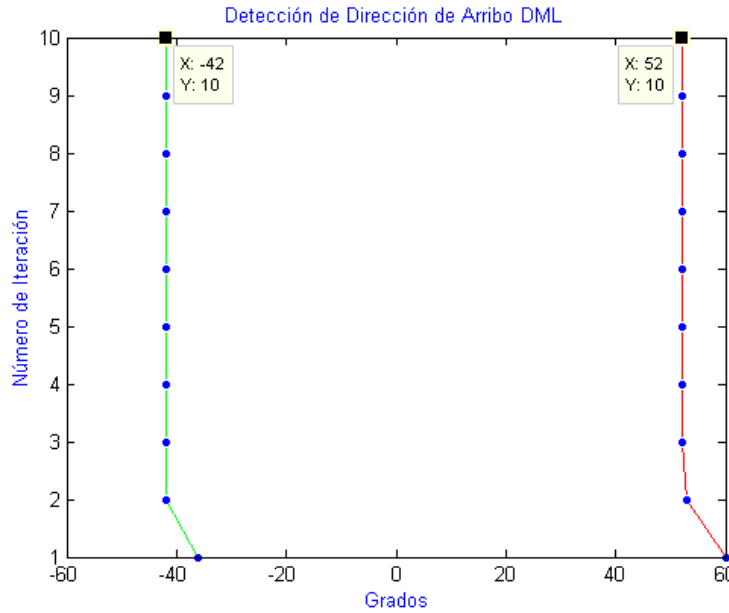


Figura 4.68: DML $\theta_1 = -37$ y $\theta_2 = 48$.

Al añadir ruido a las señales se contempla que las redes neuronales trabajan con pequeñas alteraciones, aun así estas son capaces de detectar los ángulos de arribo $\theta_1 = -37$ y $\theta_2 = 48$ como se observan en las figuras 4.65 y 4.66, aunado a esto vemos como los algoritmos MUSIC (4.67) dan pequeñas variaciones en el ángulo detectado, y la altitud de otros ángulos se refleja pero son bajos en amplitud a comparación de los ángulos de arribo, pero en DML (4.68) vemos cómo cambia mucho el ángulo de detección para ambas señales incidentes, a pesar de ello si se considera una tolerancia de $\pm 5^\circ$, estaríamos hablando que DML está sensando lo esperado en la décima iteración.

4.4.3. 2 señales Incidentes con separación intermedia

θ_1	θ_2
-7°	8°

Redes Perceptrón Multicapa

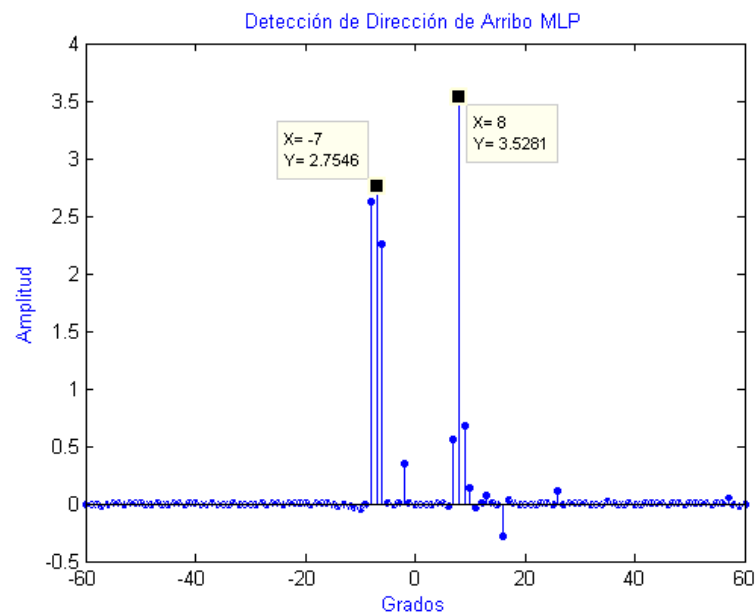


Figura 4.69: MLP $\theta_1 = -7$ y $\theta_2 = 8$.

Redes neuronales de base radial

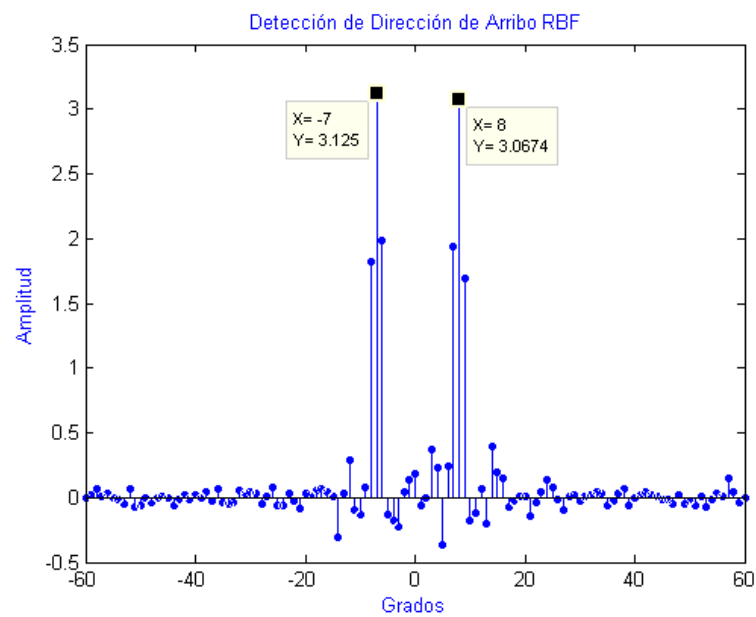


Figura 4.70: RBF $\theta_1 = -7$ y $\theta_2 = 8$.

Multiple Signal Classification

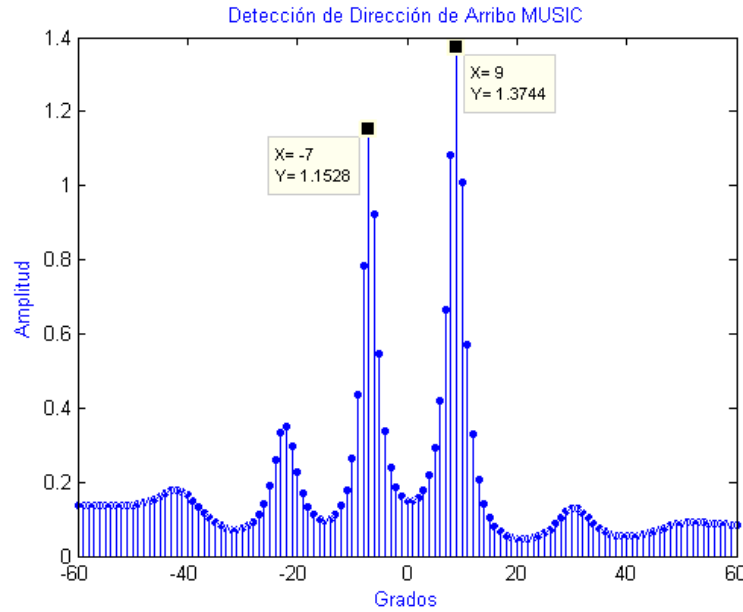


Figura 4.71: RBF $\theta_1 = -7$ y $\theta_2 = 8$.

Deterministic Maximum Likelihood

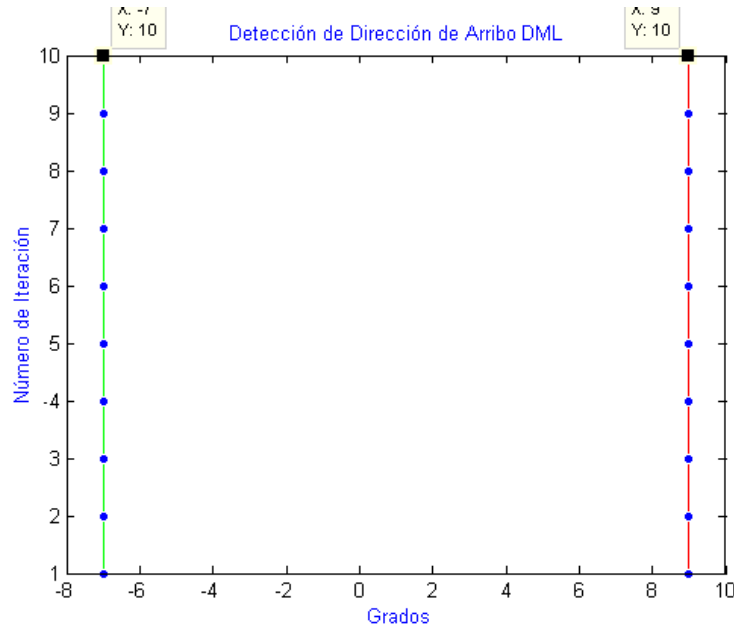


Figura 4.72: DML $\theta_1 = -7$ y $\theta_2 = 8$.

Se nota que en la red neuronal MLP (4.69) la dirección de arribo $\theta_1 = -7^\circ$ posee detecciones en los ángulos cercanos, pero viendo que ya este detalle se presentaba en casos anteriores, no puede ser objeto del aumento de ruido, igualmente tiene la mayor amplitud para los ángulos deseado, tal como ocurre con las redes neuronales RBF (4.70), además en MUSIC (4.71) vemos que el ángulo $\theta_1 = 8^\circ$ fue detectado un grado después, sin embargo se considera dentro del margen tomando una tolerancia de $\pm 1^\circ$, así mismo observa la elevación de una detección no deseada en -22° aproximadamente, pero es relativamente minoritaria con respecto a las otras dos detecciones. Para DML (4.72) se observa que el algoritmo trabaja con buen desempeño dado que los ángulos son los correctos de incertidumbre de $\pm 1^\circ$.

4.4.4. 2 señales Incidentes próximas

θ_1	θ_2
24°	25°

Redes Perceptrón Multicapa

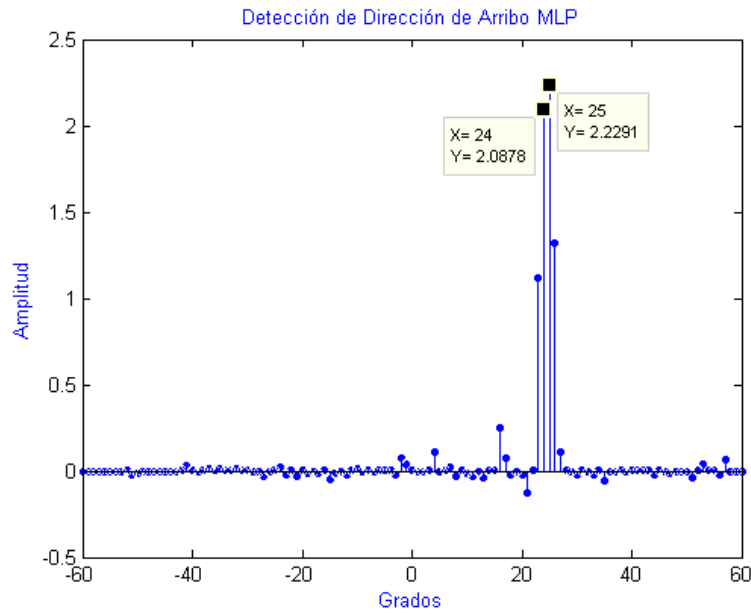


Figura 4.73: MLP $\theta_1 = 24$ y $\theta_2 = 25$.

Redes neuronales de base radial

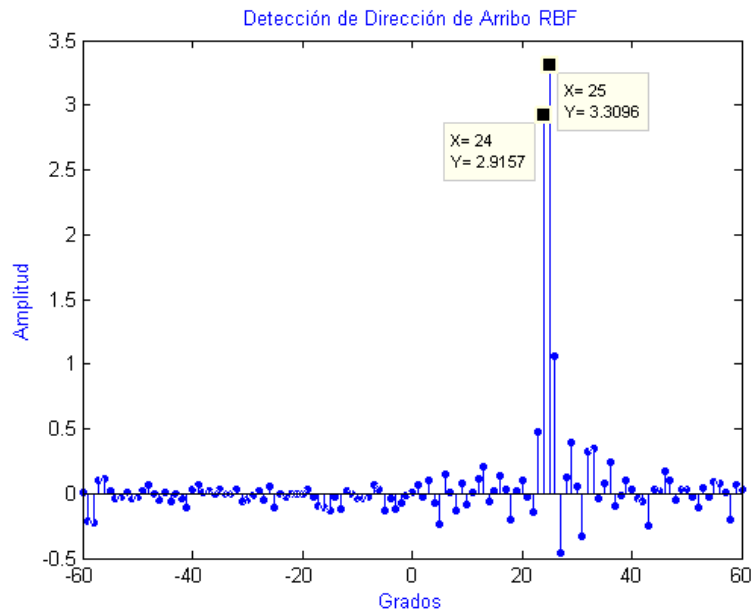


Figura 4.74: RBF $\theta_1 = 24$ y $\theta_2 = 25$.

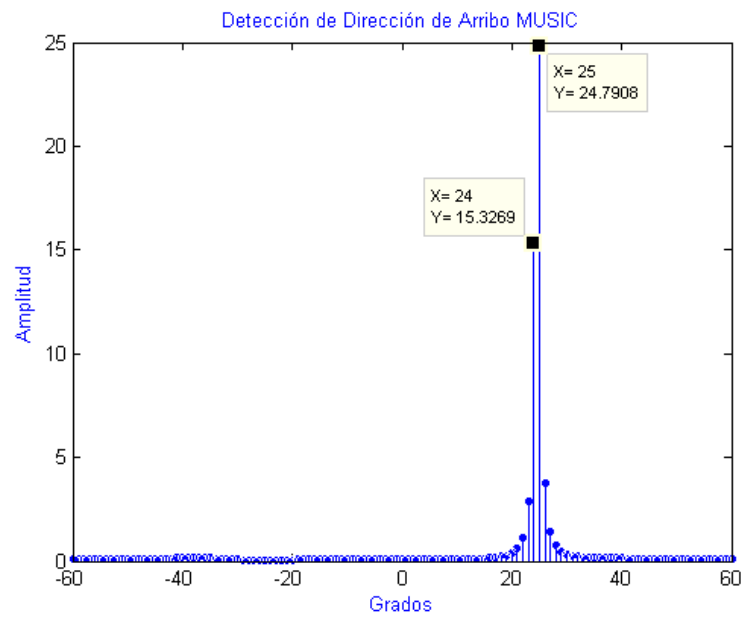
Multiple Signal Classification

Figura 4.75: MUSIC $\theta_1 = 24$ y $\theta_2 = 25$.

Deterministic Maximum Likelihood

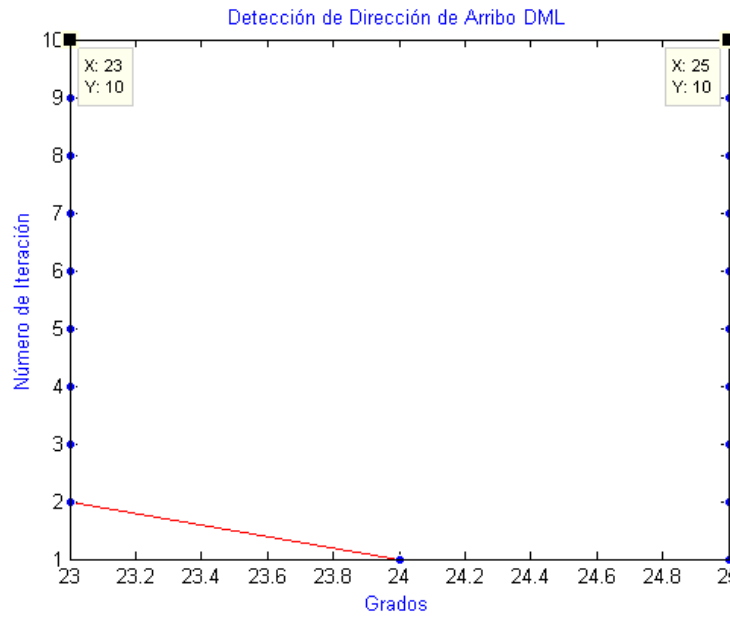


Figura 4.76: DML $\theta_1 = 24$ y $\theta_2 = 25$.

El mejor desempeño sin duda lo tiene MUSIC (4.75), seguido RBF (4.74) y luego por MLP (4.73), quienes hacen las respectivas detecciones, y junto a estas existen otras que son de amplitud un poco más bajas, para el caso MUSIC y RBF se considera desapercibidas, DML (4.76) hace detección del ángulo $\theta_2 = 25$, y del otro ángulo tiene un error de $\pm 1^\circ$ nuevamente.

4.4.5. 4 señales Incidentes

θ_1	θ_2	θ_3	θ_4
-58°	-24°	17°	58°

Redes Perceptrón Multicapa

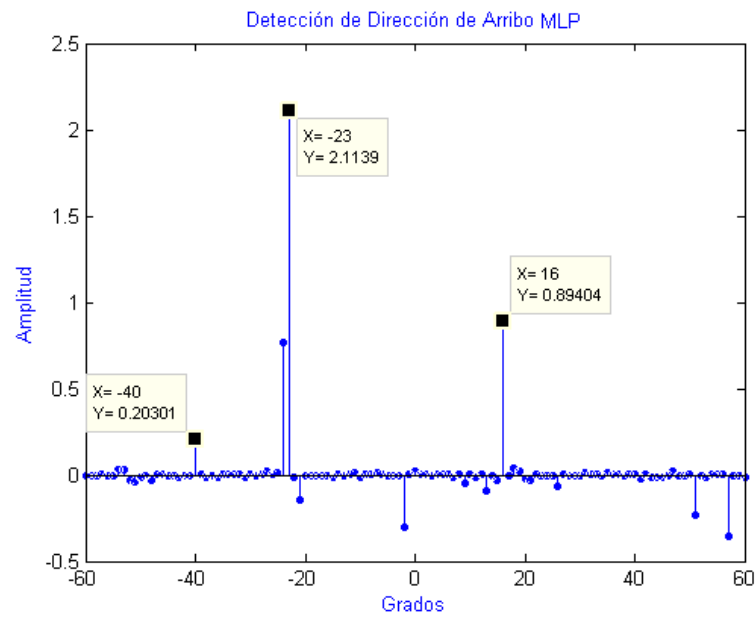


Figura 4.77: MLP $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Redes neuronales de base radial

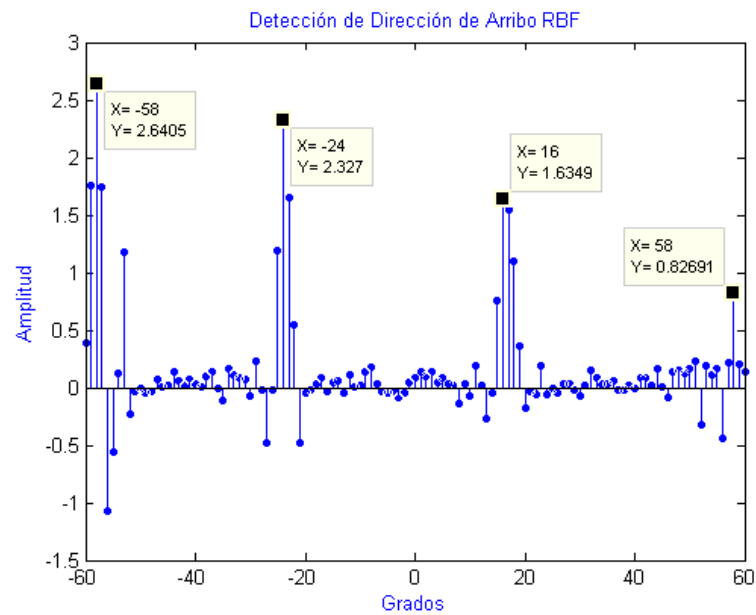


Figura 4.78: RBF $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Multiple Signal Classification

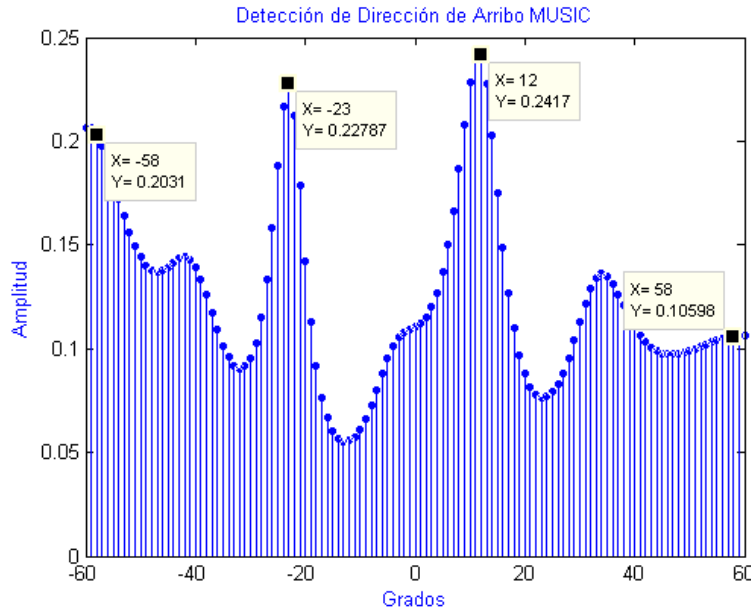


Figura 4.79: MUSIC $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Deterministic Maximum Likelihood

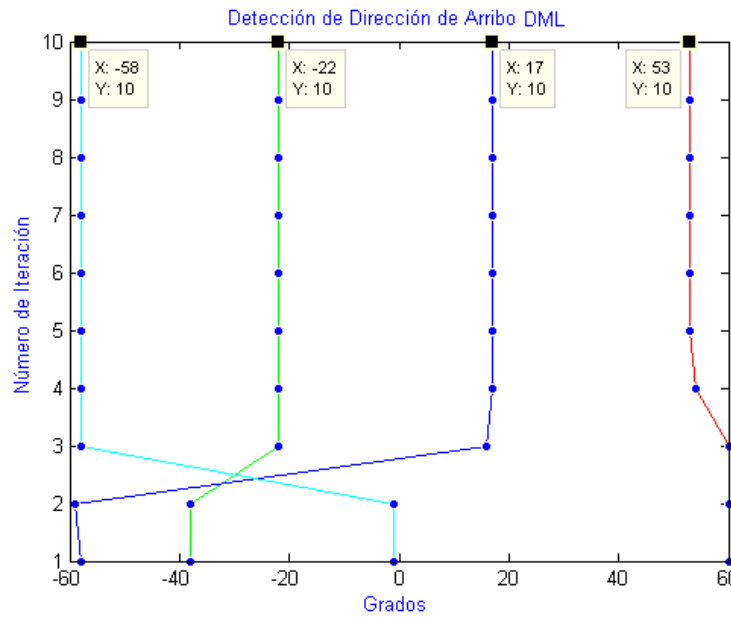


Figura 4.80: DML $\theta_1 = -58$, $\theta_2 = -24$, $\theta_3 = 17$ y $\theta_4 = 58$.

Agregando la relación señal a ruido de 5 dB vemos cómo afecta considerablemente a MLP (4.77) ya que solo hace detección de dos señales y las otras las elimina, sabiendo que la detección tiene una incertidumbre de $\pm 1^\circ$. Para RBF (4.78) vemos que la detección es precisa en la mayoría de los ángulos de arribo donde solo $\theta_3 = 17$ fue detectado en 16° . MUSIC (4.79) no hace las correctas detecciones sino para dos ángulos incidentes donde $\theta_3 = 17$ es detectado en 12° , además se crea una detección no deseada en el ángulo 36° , por otro parte DML (4.80) consigue detectar los ángulos esperados en dos ocasiones, los otros dos ángulos específicamente $\theta_2 = -24$ y $\theta_2 = 53$ se detectan en -22° y 53° respectivamente.

4.4.6. 7 señales Incidentes

θ_1	θ_{12}	θ_3	θ_4	θ_5	θ_6	θ_7
-57°	-53°	-32°	-16°	5°	22°	53°

Redes Perceptrón Multicapa

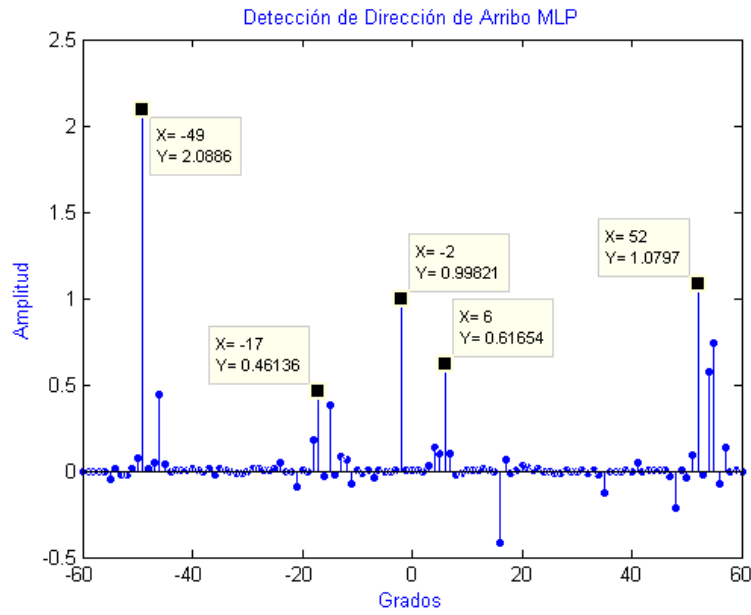


Figura 4.81: MLP $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Redes neuronales de base radial

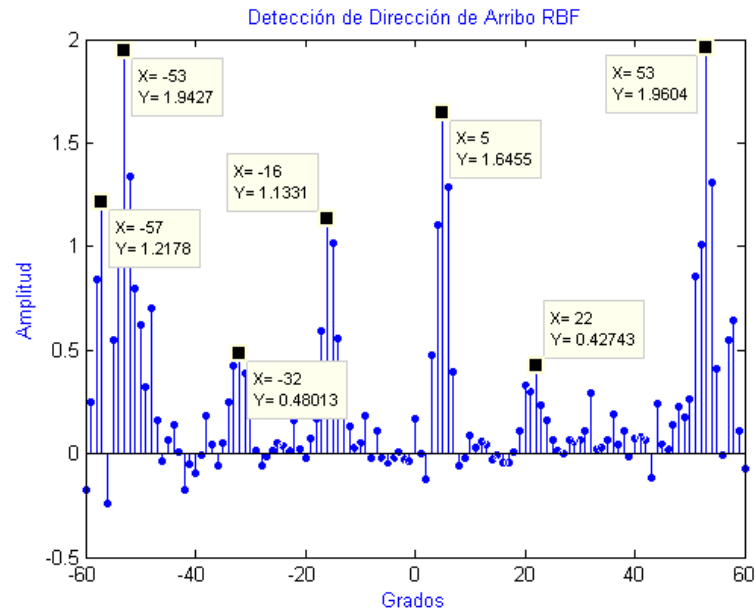


Figura 4.82: RBF $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Multiple Signal Classification

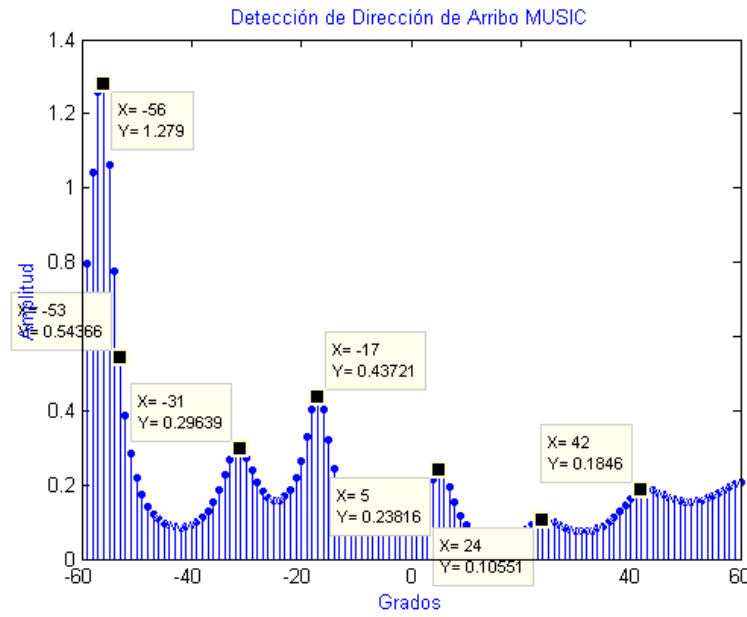


Figura 4.83: MUSIC $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Deterministic Maximum Likelihood

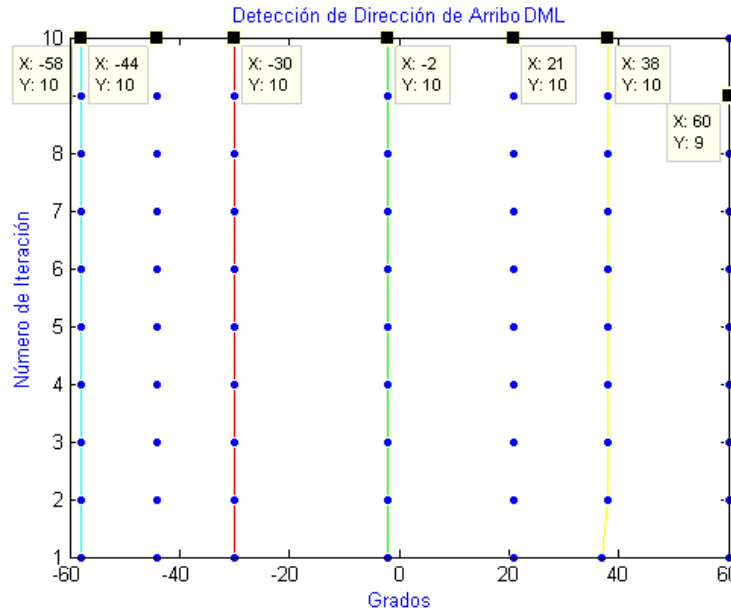


Figura 4.84: DML $\theta_1 = -57$, $\theta_2 = -53$, $\theta_3 = -32$, $\theta_4 = -16$, $\theta_5 = 5$, $\theta_6 = 22$ y $\theta_7 = 53$.

Se nota como para MLP (4.81) estas detecciones son incorrectas en su mayoría, detectando solamente $\theta_4 = -16^\circ$, $\theta_2 = -53^\circ$, $\theta_7 = 53^\circ$ y $\theta_5 = 5^\circ$ con cierto grado de error general de 4° en el peor de los casos, para RBF (4.82) la Relación Señal a Ruido se ve que no afecta del todo, a pesar de que se visualizan ciertas detecciones no favorables, por su parte en MUSIC (4.83) la amplitud detectada en $\theta_1 = -57$ es muy superior con las demás detecciones, induciendo a que las demás no sean detectadas, esto ocurre para los otros casos donde varía SNR. Nuevamente DML (4.84) tiene el mismo desempeño, ya que arroja los mismos valores que en los casos anteriores donde incidían 7 señales con distintas direcciones de llegada alejadas solo un poco de lo esperado.

Capítulo V

Conclusiones y recomendaciones

5.1. Conclusiones

El objetivo principal de cualquier sistema de comunicaciones es garantizar el servicio a cualquier abonado lo requiera, para ello se emplean diferentes y optimizan todos los recursos para lograr este cometido. El sistema de antenas inteligentes es el principal elemento a optimizar, debido a que es este el que ha brindado una mayor y mejor cobertura a los usuarios. La clave para el correcto funcionamiento de estas prácticas radica en la correcta utilización de algoritmos, tanto de detección de arribo, como de conformación de haz.

Este trabajo de grado se enfocó en el desarrollo de alternativas que generen soluciones prácticas y confiables a los métodos más comúnmente usados en la detección de dirección de arribo DOA, de señales. Los algoritmos que están a la vanguardia en la detección de arribo son MUSIC que se basa en sub-espacios, y el algoritmo DML el cual se basa en métodos determinísticos o paramétricos para calcular la dirección de llegada de cualquier señal.

Los algoritmos mencionados formaron una base objetiva en el proceso de calificar el método planteado en esta investigación, como se menciona a lo largo de todo el trabajo, la alternativa que se propone es a partir de redes neuronales configurar una estructura que sea capaz de realizar la detección de arribo de señales,

los dos tipos redes neuronales que se escogieron para realizar este ensayo fueron redes neuronales de base radial(RBF), y redes neuronales basadas en perceptrón multicapa (MLP).

Dada la configuración planteada, de una red neuronal capaz de detectar una señal en un ángulo determinado, se mostró como se comportó de manera ideal en los casos de 1 señal incidente y de 2 señales incidentes, incluso se podría decir que a los niveles de igualar el rendimiento de los algoritmos MUSIC y DML, entre las redes neuronales se observa como dependiendo del tipo tiene una respuesta diferente, aunque las dos son igualmente, eficientes y eficaces, la red MLP denota una respuesta más exacta que la red RBF, esto debido al tipo de aprendizaje que manejan las redes RBF, las cuales al ser la función de activación de tipo gaussiana, siempre tendrá una respuesta la red, a pesar de que no sea el ángulo correcto, esta respuesta se lee como ruido y es completamente diferenciable de una respuesta correcta. Las redes MLP como las RBF, tienen además un alto índice de no correccionalmente de las señales de entrada para 2 señales incidentes, siendo capaces de diferenciar con una precisión, de 1 grado de separación, característica que resalta, en cualquier algoritmo de detección de señales debido a lo importante de distinguir ángulos de incidencia con pequeña separación.

Las redes neuronales al ser entrenadas de manera correcta son capaces de detectar y distinguir, incluso, mas señales con las cuales se han entrenado por desgracia, el aprendizaje en ninguno de los casos es el suficiente como tener el mismo rendimiento, que bajo las condiciones de entrenamiento en las cuales se planteó, aunque son capaces de detectar los ángulos de llegada, no en todos los casos los detecta de manera correcta, y en determinadas circunstancias críticas, cuando el número de señales incidentes es superior a 5, se empieza a notar que el sistema puede no percibir alguna de las señales de entrada, de esta manera se puede aseverar el desempeño de la red, va relacionado directamente con la cantidad de señales con las que se puede entrenar, por desgracia como se explicó anteriormente los equipos en los cuales se desarrolló esta tesis de grado, no se podía entrenar la red con más de 2 señales de llegada, debido a limitantes de hardware.

Luego de un profundo análisis a los resultados obtenidos, se presentan las redes neuronales como una alternativa firme y confiable a la detección de señales de arribo, una alternativa que en caso de contar con los elementos necesarios, con facilidad diera atención a la cantidad de usuarios requeridos.

5.2. Recomendaciones

La realización de este trabajo de grado, se espera actué como inspiración a futuros investigadores, los cuales deseen desarrollar más este campo de las redes neuronales, para ellos se plantean ciertas consideraciones y recomendaciones:

- Desarrollar en un D.S.P. las distintas redes neuronales para evaluar su ejecución y desenvolvimiento en un entorno distinto al lenguaje de programación exclusivo
- Lograr el entrenamiento de las redes neuronales con más de 3 señales de llegada, con lo cual se podría establecer una relación más completa de cuanto entrenamiento es necesario para satisfacer las necesidades de un determinado sistema de comunicaciones inalámbrico.
- Dada la velocidad con que se desarrollan nuevas tecnologías, sería conveniente realizar una investigación de cómo se comportan las redes neuronales al manejar estándares de acceso diferentes como LTE o futuros superiores.

Apéndice A

Apendice A

1.1. Entrenamiento

A continuación se ilustran los scrips usados para la generación, evaluación y puesta en funcionamiento de las redes neuronales

% ENTRENAMIENTO

```
ss0=[]; %inicializacion de las variables a usar en la función de entrenamiento
ss=[]; %se inicializa en 0 la matriz de dirección de llegada
No=1; %numero de señales incidentes
dd=2; %distancia entre los dipolos en este caso lambda /2
Nss=10; %numero de dipolos a usar
k=100; %numero de muestras a tomar de las señales de llegada
Prr=0.001; %ruido de entrenamiento (se puede ser como se hacerca a 0)
fb=20e3; %frecuencia de la señal en banda base
fp=50e6; %frecuencia de la señal luego de la multiplexacion
f0s=200e6; %frecuencia de muestreo de la señal
for kk=2:2:121; %ciclo que crea las redes neuronales pares
ss0=[];
```

```

ss=[];

for ANG=0:1:120 ;                                %creación de la matriz de llegada de
a=(ANG-60)*pi/180;                                %las posibles respuestas de los dipòlos
for tt=1:10;
ss(tt,ANG+1)=exp(1i*2*pi*((tt-1)/dd)*(sin(a)));

if ANG==kk-1;                                     %se especifica la amplitud en
degree(ANG+1)=3;
                                                    %caso de ser el angulo al cual se entrenara

else
degree(ANG+1)=0;
end
end
end
for no=1:1:1;                                     %se entrena con una sola señal de llegada
for sam=1:1:k;                                     %creacion de las señales en banda GSM , ya multiplexada
ss0(no,sam)=cos(2*pi*fb)*cos(2*pi*sam*fp/fs);
end
end
nn=sqrt(Prr)*randn(Nss,k);                        %creacion del vector de error

for t=1:1:121;                                     %generacion de la mattriz de correlacion
Rx=ss(:,t)*ss0+nn;
Rxxx1=(Rx*conj(Rx)')/k;
be=Rxxx1(1,:);
bee=size(be);

```

```

for p=1:1:bee(2);
bb(p)= real(be(p));
bb(p+bee(2))= imag(be(p));
end
zz=bb/norm(bb);
zzmat(:,t)=zz;                                %formacion de parte del vector de entrenamiento
end
cont=0;
for ANG=0:1:120;                                %se genera nuevamente la respuesta de los dipolos
                                                %esta vez para crear las 2 señales de entrenamiento
                                                %en algunos variables

a=(ANG-60)*pi/180;
for tt=1:10;
ss(tt,ANG+1)=exp(1i*2*pi*((tt-1)/dd)*(sin(a)));
end
end
for ANG=0:1:120;
a=(ANG-60)*pi/180;
for tt=1:10;
ss1(tt,ANG+1)=exp(1i*2*pi*((tt-1)/dd)*(sin(a)));
end
end
for no=1:1:2;                                    %se entrena con 2 señales de llegada
for sam=1:1:k;
ss0(no,sam)=cos(2*pi*fb)*cos(2*pi*sam*fp/fs);
end
end
for t=2:2:121;                                    %se va corriendo de 2 en 2 todos los posibles angulos de llegada
for t1=2:2:121;                                    %hasta cubrir todas las posibles combinaciones de llegada
cont=cont+1;
ss2=[ss(:,t),ss1(:,t1)];
Rx=ss2*ss0+nn;                                    %creacion de las matriz de correlacion

```

```

Rxxx1=(Rx*conj(Rx)')/k;           %ahora con 2 señales de llegada
be=Rxxx1(1,:);
bee=size(be);
for p=1:1:bee(2);
    bb(p)= real(be(p));
    bb(p+bee(2))= imag(be(p));
end
zz=bb/norm(bb);
zzmat(:,cont+121)=zz;             %formacion de la ultima parte del vector de entrenamiento
if t==kk | t1==kk                 %en caso de que alguna señal coincida con el angulo
    degre(121+cont)=3;            %en el cual se esta entrenando se adecua la amplitud a 3
else
    degre(121+cont)=0;
end
end
end
eval(sprintf(c_d = train(c_r, zzmat, degre), kk))
                                %comando generador
                                % de las distintas redes neuronales

end

```

1.1.1. Evaluación

A continuación el código que se uso para evaluar el rendimiento de cada red

% EVALUACIÓN

```

                                %inicializacion de las variables
dd=2;                          %separación de los dipolos lambda/2
Nss=10;                         %numero de dipolos a usar

```

```

k=100;                                %numero de muestras a tomar
Prr=0.01;                             %potencia de ruido a usar en el análisis
zz2mat=[];                            %inicializacion de variables de apoyo
sss0=[];                              %inicializacion de variables de apoyo
pppi1=0;                              %inicializacion de variables de apoyo
for hh=1:10;                          %comienzo del ciclo por el cual se evalua cada red neuronal
for ANG=0:1:120;                      %creacion de la posible espuesta de los dipolos
a=(ANG-60)*pi/180;
for tt=1:10;
ss(tt,ANG+1)=exp(1i*2*pi*((tt-1)/dd)*(sin(a)));
end
end
nn=sqrt(Prr)*randn(Nss,k);            %introduccion de ruido a la señal a evaluar
for no=1:1:2;                         %generacion de las señales de llegada de tipo GSM
for sam=1:1:k;
sss0(no,sam)=cos(2*pi*fb/no)*cos(2*pi*sam*fp/fs+(200e3)*no);
end
end
prome=0;
zz2mat=[];
cont=0;
for prom=1:1:1000;                   %se calcula un promedio con 1000 ciclos distintos
                                     %para obtener un indice de acierto completo
ttt=120;
all=120*rand(1,10);

                                     %se crean 10 señales incidentes distintas para
                                     %evaluar su eficacia

t1=floor(all(1))+1;
t2=floor(all(2))+1;
t3=floor(all(3))+1;
t4=floor(all(4))+1;
t5=floor(all(5))+1;

```

```

t6=floor(all(6))+1;
t7=floor(all(7))+1;
t8=floor(all(8))+1;
t9=floor(all(9))+1;
t=22;                                %Eleccion del angulo a evaluar
cont=cont+1;

                                %Eleccion del numero de señales que se desean detectar
%ss2=[ss(:,t)];
ss2=[ss(:,t),ss1(:,t1)];            % en este caso se selecciona probar con 2 señales incidentes
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4),ss1(:,t5)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4),ss1(:,t5),ss1(:,t6)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4),ss1(:,t5),ss1(:,t6),ss1(:,t7)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4),ss1(:,t5),ss1(:,t6),ss1(:,t7),ss1(:,t8)];
%ss2=[ss(:,t),ss1(:,t1),ss1(:,t2),ss1(:,t3),ss1(:,t4),ss1(:,t5),ss1(:,t6),ss1(:,t7),ss1(:,t8),ss1(:,t9)];
Rx=ss2*sss0+nn;                    %creacion de la matriz de correlacion
Rxxx1=(Rx*conj(Rx)')/k;
be=Rxxx1(1,:);
bee=size(be);
for p=1:1:bee(2);
bb(p)= real(be(p));
bb(p+bee(2))= imag(be(p));
end
zz=bb/norm(bb);
zz2mat(:,cont)=zz;                  %formacion del vector que se introduce a las redes neuronales
end
close all
Y = sim(c22, zz2mat);               %evaluacion de las red neuronal, en este caso la 22 MLP
plot(Y)
pppi1=0;                            %toma de los valores de la respuesta de la neurona y se genera

```

```

                                %un promedio de todas las mediciones

for pp=1:1000;
if Y(pp)>1;
pppi1=pppi1+1;
end
end
pppi1;
pppi1=(pppi1+pppi1);
end
pppi1/10

```

1.2. Funcionamiento

EL siguiente código es básicamente el mismo para las arquitecturas RBF y la MLP, solo cambia la red neuronal a evaluar.

```

No=input('Ingrese el numero de señales incidentes:');
Pr=input('Ingrese la relacion SNR en dB:');
cont=0;
dd=2;                                %Divedendo de la distancia entre dipolos en este caso Lamnbda
Nss=10;                              %numero de dipolos a usar
k=100;                              %numero de muestras que se tomara de la señal de entrada
zz2mat=[];                          %inicializacion de las variables de apoyo
sss0=[];                             %inicializacion de las variables de apoyo
pipi1=0;                             %inicializacion de las variables de apoyo
ss0=[];                              %inicializacion de las variables de apoyo
fb=20e3;                             %frecuencia en banda base
fp=50e6;                             %freceuncia a la cual se multiplexa para muestrear
fs=200e6;                            %freceuncia de muestreo
for ANG=0:1:120;                     %creacion de la respuesta de las antenas a todos

```

```

                                %los posibles angulos de llegada desde -60 a 60

a=(ANG-60)*pi/180;
for tt=1:10;
ss(tt,ANG+1)=exp(1i*2*pi*((tt-1)/dd)*(sin(a)));
end
end
for no=1:1:No;                                %creacion del numero de señales en banda
                                %GSM tomando en cuenta lo introducido por el usuario
for sam=1:1:k;
sss0(no,sam)=cos(2*pi*fb/no)*cos(2*pi*sam*fp/fs+(200e3)*No);
end
end
prome=0;
zz2mat=[];
cont=0;
u=60;
for i=1:No;                                %se toman los angulos a usar directamente
                                %del usuario se recuerda que debe ser entre -60 y 60
ang(i)=input('ingrese los angulos incidentes en grados entre -60 y 60:');
t(i)=u+ang(i);
end
cont=cont+1;                                %este contador se usa para rellenar el vector que se introduce a las rec
if No==1                                %debido a que dependiendo de las necesidades
                                %del usuario varian, al igual que la matriz de correlacion
                                %es necesario colocar un IF de manera de crear la matriz de
                                %angulos incidentes adecuada

ss2=[ss(:,t(1))];
elseif No==2
ss2=[ss(:,t(1)),ss(:,t(2))];
elseif No==3
ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3))];
elseif No==4

```

```

ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3)),ss(:,t(4))];
elseif No==5
ss2=[ss(:,t(1)),ss1(:,t(2)),ss1(:,t(3)),ss1(:,t(4)),ss1(:,t(5))];
elseif No==6
ss2=[ss(:,t(1)),ss1(:,t(2)),ss1(:,t(3)),ss1(:,t(4)),ss1(:,t(5)),ss1(:,t(6))];
elseif No==7
ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3)),ss(:,t(4)),ss(:,t(5)),ss(:,t(6)),ss(:,t(7))];
elseif No==8
ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3)),ss(:,t(4)),ss(:,t(5)),ss(:,t(6)),ss(:,t(7)),ss(:,t(8))];
elseif No==9
ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3)),ss(:,t(4)),ss(:,t(5)),ss(:,t(6)),ss(:,t(7)),ss(:,t(8)),ss(:,t(9))];
elseif No==10
ss2=[ss(:,t(1)),ss(:,t(2)),ss(:,t(3)),ss(:,t(4)),ss(:,t(5)),ss(:,t(6)),ss(:,t(7)),ss(:,t(8)),ss(:,t(9)),ss(:,t(10))];
end
Rx=ss2*sss0; %multiplicacion de las señales incidentes con
               %la respuesta de cada antena
Rx = awgn(Rx,Pr,'measured'); %introduccion de SNR a la señal total de llegada
Rxxx1=(Rx*conj(Rx)')/k; %calculo de la matriz de correlacion
be=Rxxx1(1,:); %se toma la primera fila de esta matriz que la que
               %contiene la informacion necesaria
bee=size(be);
for p=1:1:bee(2); %este ciclo permite generar separa y normalizar
                  %las distintas vestores de la matriz de correlacion
bb(p)= real(be(p));
bb(p+bee(2))= imag(be(p));
end
zz=bb/norm(bb); %normalizacion del vector de entrada
zz2mat(:,cont)=zz; %creacion del vector que se introduce a la red neuronal
close all
               %introduccion del vector fila de la correlación
               %en todas las redes neuronales para su
               %respectiva evaluacion por grados

```

```
Y(1) = sim(net1, zz2mat);
Y(2) = sim(net2, zz2mat);
:
Y(120) = sim(net120, zz2mat);
Y(121) = sim(net121, zz2mat);
ph=stem([-60:1:60],Y); %grafico de la respuesta de las redes neuronales RBF
title('Deteccion de Direccion de Arribo')
xlabel('Grados')
ylabel('Amplitud')
set(ph,'marker','o')
set(ph,'markersize',5)
set(ph,'markerfacecolor','b')
set(ph,'markeredgecolor','w')
```

Apéndice B

Apendice B

2.1. Algoritmo MUSIC

```
No=input('Ingrese el numero de señales incidentes:');
Pr=input('Ingrese la relacion SNR en dB:');
Ns=10;                                %numero de dipolos a usar
k=100;                                %numero de muestras tomadas de la señal
d=2;                                  %Divedendo de la distancia entre dipolos en este caso Lamnbda
ang(No)=0;                            %inicializacion de las señales de apoyo
for i=1:No;                            %introduccion por parte del usuario de los grados de llegada
    ang(i)=input('ingrese los angulos incidentes en grados:');
    angrad(i)=ang(i)*pi/180;
end
fb=20e3;                              %frecuencia en banda base
fp=50e6;                              %frecuencia de multiplexacion a la cual se toman las muestras
fs=200e6;                             %frecuencia de muestreo
s(Ns,No)=0;                           %inicializacion de la matriz de angulos de llegada
for t=1:Ns;
    for r=1:No;
        s(t,r)=exp(1i*2*pi*((t-1)/d)*(sin(angrad(r))));
    end
end
```

%llenado de la matriz

```

                                %de respuesta de los dipolos
end
end
for no=1:1:No                                %creacion de las señales de llegada en la
                                                banda de GSM multiplexada a 50MHz
for sam=1:1:k;
s0(no,sam)=cos(2*pi*fb)*cos(2*pi*sam*((fp+no*200e3)/fs));
end
end
c=s*s0;                                %creacion de la señal completa de llegada
Rx = awgn(c,Pr,'measured');            %introduccion del SNR a la señal de llegada
Rxx=(Rx*conj(Rx)')/k;                  %Formacion de la matriz de correlacion
[avect aval]=eig(Rxx);                  %autovectores y autovalores de la matriz de correlacion
minevalue=abs(aval(1,1));               %inicializacion del valor minimo
for u=2:Ns                                %busqueda de los autovectores asociados a ruido,
y autovectores asociados a señal
if(minevalue>abs(aval(u,u)));
minevalue=abs(aval(u,u));
end
end
z=0
for y=1:Ns;
if(aval(y,y)<(5*minevalue)+ minevalue);
z=z+1;
end
end
autruid(Ns,z)=0;
for q=(Ns-z)+1:Ns
for L=1:Ns
autruid(L,q-(Ns-z))=avect(L,q);
end
end
end

```

```

cont=1;
for tet=-60:1:60;                                %busqueda del pico maximo de respuesta donde
                                                    %el agoritmo music detecta el angulo de la
                                                    %señal de llegada desde -60 a 60

hh(cont)=tet;
radian=tet*pi/180;
for w=1:Ns
matsen(w,1)=exp(1i*2*pi*((w-1)/d)*(sin(radian)));
end
ah=matsen'*autruid;
je=ah*ah';
pmu(cont)=1/abs(je);
cont=cont+1;
end
degre3=60:-1:-60;
ph=stem(degre3,pmu)                                %grafica de los resultados obtenidos del algoritmo MUSIC
title('Deteccion de Direccion de Arribo MUSIC','color','b')
xlabel('Grados','color','b')
ylabel('Amplitud','color','b')
set(ph,'marker','o')
set(ph,'markersize',5)
set(ph,'markerfacecolor','b')
set(ph,'markeredgecolor','w')

```

2.2. Algoritmo DML

```

No=input('Ingrese el numero de señales incidentes:');
Pr=input('Ingrese la relacion SNR en dB:');
Ns=10;                                              %numero de dipolos a usar
k=100;                                              %numero de muestras tomadas de la señal

```

```

d=2; %Divedendo de la distancia entre dipolos en este caso Lamnbda /2
ang(No)=0; %inicializacion de las señales de apoyo
for i=1:No; %introduccion por parte del usuario
    de los grados introducidos por el usuario
    ang(i)=input('ingrese los angulos incidentes en grados:');
    angrad(i)=ang(i)*pi/180;
end
fb=20e3; %frecuencia en banda base
fp=50e6; %frecuencia de multiplexacion a la cual se toman las muestras
fs=200e6; %frecuencia de muestreo
s(Ns,No)=0; %inicializacion de la matriz de angulos de llegada
for t=1:Ns;
    for r=1:No;
        s(t,r)=exp(1i*2*pi*((t-1)/d)*(sin(angrad(r))));
        %llenado de la matriz de respuesta de los dipolos
    end
end
for no=1:1:No %creación de las señales de llegada
    %en la banda de GSM multiplexada a 50Mz
    for sam=1:1:k;
        s0(no,sam)=cos(2*pi*fb)*cos(2*pi*sam*((fp+no*200e3)/fs));
    end
end
c=s*s0; %creacion de la señal completa de llegada
Rx = awgn(c,Pr,'measured'); %introduccion del SNR a la señal de llegada
Rxx=(Rx*conj(Rx)')/k; %Formacion de la matriz de correlacion
ejecucion=5; %numero de veces que se ejecutara el programa
D = complex(single(zeros(10,5))); %inicializacion de las variables de
    creación de los valores
PiA = complex(single(zeros(10,10)));%inicializacion de las variables de la diagonal
    de la matriz de correlación
Valor = complex(single(zeros(1,1))); %inicializacion de las variables del valor menor

```

```

THETADML = single(zeros(1,5)); %inicializacion de las variables de salida
n = 1; %inicializacion de las variables de apoyo
a1 = single(zeros(1,1)); %inicializacion de la variable de apoyo
                                %de resolución de angulos

while n <= ejecucion %comienzo del ciclo de deteccion del angulo
for i = 1:No
menor = single(1000000); %inicializacion de la variable de control "menor"
for pasos = single(-60:1:60) %establecimiento de los pasos por los cuales
                                %se evaluara el angulo de arribo

for l = 1:10
D(l,i) = exp(1i*2*pi*((l-1)/d)*(sin(pasos*pi/180)));
                                %formacion de la matriz de respuesta de los dipolos
end
PiA = D*pinv(D); %multiplicación de la matriz de respuesta de
los dipolos por su inversa
Proy = eye(size(PiA)) - PiA;
Valor = (trace(Proy*(Rxx))); %se obtiene la traza de
                                % la matriz de la correlación de la señal de llegada
                                % multiplicado por la proyección de la matriz de respuesta
                                %de los dipolos

if abs(Valor) < menor
menor = abs(Valor); %en caso de que la respuesta sea un
                                %numero muy bajo se toma como posible angulo de
                                %llegada

a1 = pasos;
end
end
THETADML(i,n) = a1
for l = 1:10 %se reescribe la matriz de respuesta de los dipolos
                                % tomando en cuenta el angulo de la tendencia
                                %encontrada
D(l,i) = exp(1i*2*pi*((l-1)/d)*(sin(a1*pi/180)));

```

```
end
end
n = n + 1 ;
end

                                %grafico de la respuesta encontrada
ph=plot(THETADML(1,:),[1:5], 'r', THETADML(2,:),[1:5], 'g', THETADML(3,:),[1:5], 'o')
THETADML(4,:),[1:5], 'm', THETADML(5,:),[1:5], 'r', THETADML(6,:),[1:5], 'k',)
THETADML(7,:),[1:5], 'c');
title('Deteccion de Direccion de Arribo DML', 'color', 'b')
xlabel('Grados', 'color', 'b')
ylabel('Numero de Iteración', 'color', 'b')
set(ph, 'marker', 'o')
set(ph, 'markersize', 5)
set(ph, 'markerfacecolor', 'b')
set(ph, 'markeredgecolor', 'w')
```

Referencias Bibliográficas

- [1] Amevir Castillo and Elimes Rodriguez. Simulacion y estudio comparativo de los algoritmos de deteccion de direccion de arribo (doa), usado en antenas inteligentes. Master's thesis, Universidad de Carabobo, 2005. URL [http://produccion-uc.bc.uc.edu.ve/cgi-bin/wwwisis/\[in=fichero/fichero.in\]](http://produccion-uc.bc.uc.edu.ve/cgi-bin/wwwisis/[in=fichero/fichero.in]).
- [2] Borja Errasti Alcala. Estimación de dirección de llegada basada en métodos de optimización metaheurística mediante un Único muestreo. Master's thesis, Universidad de Navarra, 2012. URL http://www.uah.es/estudios/postgrado/detalle_proyectos.asp?CdPlan=D326&expe=8&proy=1.
- [3] Villanueva Espinoza and Maria del Rosario. Las redes neuronales artificiales y su importancia como herramienta en la toma de decisiones. Master's thesis, Universidad Nacional Mayor de San Marcos, 2002. URL http://sisbib.unmsm.edu.pe/bibvirtual/tesis/Basic/Villanueva_EM/villanueva_em.htm.
- [4] Ortiz Contreras, Bolívar José, Arévalo Encalada, and Wilmer Cristian. Diseño e implementación de un simulador para analizar el funcionamiento de un sistema de antenas inteligentes. Master's thesis, Universidad Politécnica Salesiana, 2009. URL <http://dspace.ups.edu.ec/handle/123456789/15>.
- [5] Ocampo Jonathan. Reconocimiento de caracteres de una placa de automóvil mediante redes neuronales artificiales utilizando matlab. Master's thesis, Escuela Politecnica del Ejercito, 2011. URL <http://repositorio.espe.edu.ec/bitstream/21000/4939/1/T-ESPE-033009.pdf>.

- [6] Alberto Hernandez. Analisis comparativo de algoritmos de direccion de llegada basados en subespacios para antenas inteligentes music y spirit orientados a redes moviles 3g. Master's thesis, Universidad Austral de Chile, 2010. URL <http://cybertesis.uach.cl/tesis/uach/2010/bmfcih557a/doc/bmfcih557a.pdf>.
- [7] Robles Durazno I. and Sarmiento Durán G. Calidad de las operadoras celulares en cuenca. Master's thesis, Universidad Politécnica Salesiana, 2007. URL <http://dspace.ups.edu.ec/bitstream/123456789/177/1/Indice.pdf>.
- [8] Stevanović I., Skrivervik A., and Mosig J. Smart antenna systems for mobile communications. Master's thesis, Laboratoire d'Electromagnétisme et d'Acoustique Ecole Polytechnique Fédérale de Lausanne, 2003. URL http://infoscience.epfl.ch/record/140902/files/smart_antennas.pdf.
- [9] Ortiz Contreras, Bolívar José, Arévalo Encalada, and Wilmer Cristian. Diseño e implementación de un simulador para analizar el funcionamiento de un sistema de antenas inteligentes. Master's thesis, Universidad Politécnica Salesiana, 2009. URL <http://dspace.ups.edu.ec/handle/123456789/15>.
- [10] Moreno Jiménez Óscar, Franco Martín Ignacio Javier, and Miranda Sierra José Luis. Introducción a la tecnología de antenas inteligentes, aplicación a umts. *Comunicaciones de Telefónica I+D*, 21:41–56, 2001.
- [11] Krim H. and Viberg M. Two decades of array signal processing research: The parametric approach. *IEEE Signal Processing Magazine*, 13:67–94, 1996.
- [12] Yáñez García and Ángel Miguel. Estimación de la dirección de arribo de señales a arreglos de antenas de fase. Master's thesis, Instituto Politécnico Nacional, 2013. URL <http://tesis.ipn.mx/jspui/bitstream/123456789/12050/1/ESTIMACIONDIREC.pdf>.
- [13] Panagiotis Tsakalides. *Array Signal Processing with Alpha-Stable Distributions*. PhD thesis, University of Southern California, 1995. URL <http://users.ics.forth.gr/tsakalid>.

- [14] Strang G. *Algebra Lineal y sus Aplicaciones*. Fondo Educativo Interamericano, México, 1982.
- [15] Freeman James A and Skapura David M. *Neural Networks Algorithms, Applications and Programming Techniques*. Addison-Wesley, Houston, Texas, USA, 1991.
- [16] Piedra Nelson and López Jorge. Estudio de la aplicación de redes neuronales artificiales en la ingeniería de tráfico de internet. *Universidad Politécnica de Madrid*, 2007. URL <https://nopiedra.files.wordpress.com/2007/10/aplicacion-de-redes-neuronales-artificiales-para-la-ingenieria-de-trafico-de-internet.pdf>.
- [17] L. Flórez and Fernández F. *Las Redes Neurales Artificiales Fundamentos Teóricos y Aplicaciones Prácticas*. Netbiblo, España, 2008.
- [18] De la Fuente Aparicio María Jesús and Calonge Cano Teodoro. *Aplicaciones de las Redes de Neuronas en Supervisión, Diagnosis y Control de Procesos*. Equinoccio, Caracas, Venezuela, 1999.
- [19] Delgado M., Pegalajar M.C., and Cuéllar M.P. Evolutionary training for dynamical recurrent neural networks. *Mathware y Soft Computing*, 13:89–110, 2006.
- [20] Moreno Rodríguez Alfonso. Desarrollo de una interfaz gráfica de redes neuronales usando matlab. Master's thesis, Universidad Carlos III Madrid, 2008. URL <http://e-archivo.uc3m.es/bitstream/handle/10016/8488/Proyecto%20Redes%20neuronales%20GUI.pdf?sequence=1>.
- [21] Rangel David and Maduro Jose. Simulación y evaluación de algoritmos de dirección de arribo (doa) en distintos procesadores digitales de señales (dsp's) usados en antenas inteligentes. Master's thesis, Universidad de Carabobo, 2015.