

# **UNIVERSIDAD AUTÓNOMA DE ZACATECAS**

## **UNIDAD ACADÉMICA DE ESTUDIOS NUCLEARES**



### **DETERMINACIÓN DEL ESPECTRO DE NEUTRONES MEDIANTE REDES NEURONALES ARTIFICIALES EN CPU Y GPU**

**I.C.E. Oscar Ernesto Alonso Muñoz**

Tesis de Maestría  
presentada a la Unidad Académica de Estudios Nucleares  
de acuerdo a los requerimientos de la Universidad para obtener el Grado de

### **MAESTRO EN CIENCIAS NUCLEARES CON ORIENTACIÓN EN INSTRUMENTACIÓN NUCLEAR**

Directores de tesis:  
M. en C. Víctor Martín Hernández Dávila y Dr. Héctor René Vega Carrillo

Zacatecas, Zac., México. Febrero 2017



# **Determinación del espectro de neutrones mediante redes neuronales artificiales en CPU y GPU.**

Aprobada por el jurado:

---

Dr. Héctor René Vega Carrillo

---

Dra. Consuelo Letechipía de León

---

Dr. Valentín Badillo Almaraz

---

Dr. Miguel Ángel Salas Luévano

---

MITC. Salvador Ibarra Delgado

## **Agradecimientos**

A la Universidad Autónoma de Zacatecas y a la Unidad Académica de Estudios Nucleares por permitir desarrollarme profesionalmente.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por la beca nacional y beca mixta, otorgadas durante los estudios de maestría.

A mis asesores el M. en C. Víctor Martín Hernández Dávila y el Dr. Héctor René Vega Carrillo, por todo su apoyo, motivación y por compartirme sus conocimientos.

Al Dr. Eduardo Gallego Díaz, Alfredo Lorente Fillol y Sviatovslav Ibáñez Fernández, del Departamento de Ingeniería Nuclear de la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Politécnica de Madrid. A Roberto Méndez Villafañe del Laboratorio de Metrología de Radiaciones Ionizantes del Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas en Madrid, España. Por haberme aceptado y apoyado durante la estancia de investigación realizada en las dos instituciones.

A mis compañeros Karen Guzmán y Alberto de la Torre de la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Politécnica de Madrid, por el apoyo brindado durante la estancia de investigación y en el trabajo de tesis.

A todos mis maestros por sus enseñanzas y por el tiempo que me dedicaron.

## RESUMEN

El espectro de neutrones se extiende en varias energías, por lo que el contador empleado es el Espectrómetro de Esferas Bonner (BSS), al utilizar las tasas de conteo y las Redes Neuronales Artificiales (ANNs), demuestran ser un método alternativo en la espectrometría neutrónica. La CPU está limitada a realizar cálculos computacionalmente intensivos. Por lo que una Unidad de Procesamiento Gráfico (GPU) es atractiva para la computación con ANN, ya que trabaja en paralelo. Este estudio determinó el espectro de neutrones a partir de las 7 tasas de conteo obtenidas del BSS mediante una ANN entrenada en CPU y GPU NVIDIA®. De la base de datos del Organismo Internacional de Energía Atómica (OIEA) se obtuvieron espectros de neutrones. Las tasas de conteo del BSS y el espectro están relacionados a través de la ecuación de Fredholm que es un sistema mal condicionado. Para la solución del problema se diseñó una ANN *feedforward*, conformada por 7 entradas, 2 capas ocultas y una de salida de 25, 25 y 27 neuronas. Para el entrenamiento de la red se tomaron 182 espectros, mediante el algoritmo de gradiente conjugado descendente (SCG) se actualizaron los valores de los pesos sinápticos y *bias*. Para la validación se tomaron los 12 de espectros restantes y se compararon los espectros reconstruidos por la ANN con los originales usando la prueba estadística Chi Cuadrada  $\chi^2$ . El diseño fue realizado con los *toolbox* de redes neuronales y de computación paralela, de MATLAB® 2015a. El entrenamiento se realizó en CPU con uno y varios núcleos, en CPU con GPU, y en GPU. El rendimiento computacional de las ANN es mejor con el algoritmo SCG, pero por el contrario se necesita mayor capacidad de memoria. El cuello de botella en el procesamiento entre CPU y GPU es la velocidad de transmisión en el ducto PCI-E.

**Palabras clave:** espectro de neutrones, esferas Bonner, redes neuronales artificiales, GPU, MATLAB.

## ABSTRACT

The neutron spectrum extends at several energies, so the counter used is the Bonner Spheres Spectrometer (BSS), using counting rates and Artificial Neural Networks (ANNs), prove to be an alternative method in neutron spectrometry. The CPU is limited to computationally intensive calculations. So a Graphics Processing Unit (GPU) is attractive for computing with ANN, since it works in parallel. This study determined the neutron spectrum from the 7 counting rates obtained from the BSS using an ANN-trained CPU and NVIDIA® GPU. Neutron spectra were obtained from the International Atomic Energy Agency (IAEA) database. The counting rates of the BSS and the spectrum are related through the Fredholm equation which is a poorly conditioned system. To solve the problem an ANN feedforward was designed, consisting of 7 inputs, 2 hidden layers and an output of 25, 25 and 27 neurons. For the network training 182 spectra were taken, the values of the synaptic and bias weights were updated using the gradient conjugate descending algorithm (SCG). For the validation the remaining 12 spectra were taken and the spectra reconstructed by the ANN with the originals were compared using the Chi Square test  $\chi^2$ . The design was done with the neural network and parallel computing toolbox, MATLAB® 2015a. The training was performed in CPU with one and several cores, in CPU with GPU, and in GPU. The computational performance of the ANNs is better with the SCG algorithm, but on the contrary, it requires more memory capacity. The bottleneck in processing between CPU and GPU is the transmission speed in the PCI-E duct.

**Keywords:** neutron spectrum, Bonner spheres, artificial neural networks, GPU, MATLAB.

# ÍNDICE GENERAL

Agradecimientos.....	ii
RESUMEN .....	iii
ABSTRACT .....	iv
ÍNDICE GENERAL .....	v
ÍNDICE DE FIGURAS .....	vii
ÍNDICE DE TABLAS Y ECUACIONES.....	ix
I INTRODUCCIÓN.....	1
Problema Científico .....	5
II REVISIÓN DE LITERATURA .....	6
2.1 Neutrones.....	6
Interacción de neutrones con la materia.....	8
Producción de neutrones.....	9
Neutrones a partir de reacciones ( $\alpha, n$ ) .....	10
Neutrones a partir de fotodesintegración ( $\gamma, n$ ) .....	10
Neutrones a partir de aceleradores de partículas.....	11
Neutrones a partir de reacciones de extracción .....	11
Neutrones a partir de protones de alta energía .....	12
Neutrones a partir de reactores de fisión nuclear .....	12
2.2 Detección de neutrones.....	13
Sistema espectrométrico de esferas Bonner (BSS) .....	13
Medición con el BSS .....	16
Matriz de respuesta del detector .....	20
2.3 Redes neuronales artificiales .....	21
Modelo neuronal biológico.....	23
Modelo matemático .....	25

Arquitectura .....	29
Red <i>Feedforward</i> .....	29
Proceso de aprendizaje.....	31
Algoritmo de entrenamiento SCG.....	33
2.4 GPU.....	34
Modelos de programación .....	35
Arquitectura CUDA .....	36
2.5 MATLAB® .....	39
Parallel Computing Toolbox™, .....	41
III MATERIALES Y MÉTODOS .....	43
3.1 Obtención de las tasas de conteo en la UPM y el CIEMAT.....	43
3.2 Diseño de la red neuronal artificial ANN.....	48
Entrenamiento .....	49
Implementación en MATLAB® con CPU.....	50
Implementación en MATLAB® con GPU.....	51
3.3 Validación y pruebas de la ANN.....	54
IV RESULTADOS Y DISCUSIÓN .....	55
4.1 Tasas de conteo obtenidas en la UPM y el CIEMAT.....	55
4.2 Diseño y entrenamiento de la ANN en CPU y GPU .....	58
4.3 Validación y pruebas de la ANN.....	65
CONCLUSIONES.....	78
REFERENCIAS.....	80
ANEXO A .....	85
ANEXO B .....	87
ANEXO C .....	89
ANEXO D .....	90
CURRICULUM VITAE .....	91

## ÍNDICE DE FIGURAS

Figura 1. Conjunto de moderadores del BSS (Cruzate et al., 2009). .....	14
Figura 2. Esquema de contador de neutrones (Aza et al., 2016). .....	15
Figura 3. Esquema de posibles pistas de neutrones en dos esferas Bonner de diferentes tamaños (Decker et al., 2015, Knoll, 2000). .....	17
Figura 4. Dependencia energética de las eficiencias de detección relativas del BSS (Knoll, 2000). .....	18
Figura 5. Modelo de una neurona biológica y su conexión (Chow and Cho, 2007). .....	24
Figura 6. Modelo matemático de una neurona artificial (Hernandez-Davila et al., 2014). .....	25
Figura 7. Diferentes tipos de funciones de activación (Hernandez-Davila et al., 2014). .....	27
Figura 8. Operación de una neurona simple. ....	28
Figura 9. Red feedforward con una capa oculta (Chow and Cho, 2007). .....	31
Figura 10. Modelo de programación CUDA GPU (Mei et al., 2014). .....	38
Figura 11. Esfera Bonner de 10" sobre la bancada en LMN de la UPM. ....	45
Figura 12. Esfera de 12 pulgadas con cono de sombra, en el LPN del CIEMAT. .	47
Figura 13. Comportamiento tasas de conteo de diferentes fuentes de neutrones.	57
Figura 14. GUI ntraintool en el entrenamiento de computación paralela. ....	61
Figura 15. Chequeo de validaciones. ....	62
Figura 16. Rendimiento de la red en GPU. ....	63
Figura 17. Regresión de datos durante el entrenamiento en GPU. ....	64
Figura 18. Espectros de neutrones de referencia y por ANN de ISO D <sub>2</sub> O/ <sup>252</sup> Cf. ....	66
Figura 19. Espectros de neutrones de referencia y por ANN de <sup>241</sup> AmBe PTB Cono. ....	67
Figura 20. Espectros de neutrones de referencia y por ANN de <sup>238</sup> PuBe CERN. .	68

Figura 21. Espectros de neutrones de referencia y por ANN de $^{238}\text{PuF}$ PNL a 50cm. ....	69
Figura 22. Espectros de neutrones de referencia y por ANN de Trawsfynydd Filtro. ....	70
Figura 23. Espectros de neutrones de referencia y por ANN de Hinkley Punto. ...	71
Figura 24. Espectros de neutrones de referencia y por ANN de CZECH por encima del equipo.....	72
Figura 25. Espectros de neutrones de referencia y por ANN de NTL111 a 115 cm. ....	73
Figura 26. Espectros de neutrones de referencia y por ANN de Pollux Arriba. ....	74
Figura 27. Espectros de neutrones de referencia y por ANN de PuPlant Loc5. ....	75
Figura 28. Espectros de neutrones de referencia y por ANN de $^{244}\text{Cm}$ Glovebox. 76	
Figura 29. Espectros de neutrones de referencia y por ANN de Acelerador de Boro Epidermal. ....	77

## ÍNDICE DE TABLAS Y ECUACIONES

Tabla 1. Clasificación de los neutrones por sus energías. ....	8
Tabla 2. Principales neutrones inducidos por reacciones nucleares. ....	9
Tabla 3. Constantes de proporcionalidad calculadas. ....	19
Tabla 4. Tasa de conteo de las fuentes de neutrones de la UPM. ....	55
Tabla 5. Tasa de conteo de la fuente de neutrones <sup>252</sup> Cf del CIEMAT. ....	56
Tabla 6. Rendimiento de las arquitecturas Pentium y GTX 960. ....	58
Tabla 7. Rendimiento de las arquitecturas Intel y GT 720. ....	59
Tabla 8. Resultados de $\chi^2$ para los espectros de validación. ....	65
$CD = E_{min}E_{max}R(D, E)\varphi_{EndEn} \dots\dots\dots 1 \dots\dots\dots$	20
$C_j = i = 1mR_i, j\varphi_i, m = 1, 2, 3, \dots, \text{número de esfera} \dots\dots\dots 2 \dots\dots\dots$	20
$uk = j = 1mWkjX_j \dots\dots\dots 3 \dots\dots\dots$	27
$yk = \varphi uk + bk \dots\dots\dots 4 \dots\dots\dots$	27
$\chi^2 = (E_{ANN} - E_{original})^2 E_{original} \dots\dots\dots 5 \dots\dots\dots$	54

# I INTRODUCCIÓN

Los neutrones son partículas subatómicas dentro del núcleo atómico y son producidos por la fisión y fusión nuclear entre las interacciones de los rayos cósmicos y la materia que se encuentra en la atmósfera, por fuentes isotópicas, reactores nucleares y aceleradores de partículas. El espectro de neutrones se extiende en varias energías y estas dependen de varios factores, como el material del blanco, blindaje, geometría, espesor y el ángulo de emisión, por lo tanto, la caracterización del espectro de neutrones proporciona información útil para definir la intensidad de la fuente y así desplegar el espectro de energía de neutrones (Hernandez-Davila et al., 2014, Ortiz-Rodríguez et al., 2014, Agosteo et al., 2012).

De los contadores de neutrones que existen, algunos pueden usarse sobre un rango limitado de energías mientras que otros tienen una eficiencia casi constante. El instrumento empleado normalmente para determinar las distribuciones de energía es el Espectrómetro de Esferas Bonner (BSS), introducido por “Bramblett, Ewing y Bonner” en 1960 (Bramblett et al., 1960, Agosteo et al., 2012).

El BSS es un conjunto de esferas de polietileno de alta densidad con un detector para neutrones térmicos que se encuentra en el centro de las esferas. Tiene las ventajas de respuesta isotrópica, buena discriminación de rayos gamma y la capacidad de intercambiar fácilmente cada esfera, las cuales utilizan el principio de moderación de partículas, de tal forma que permite al BSS medir el flujo de neutrones a través de un amplio rango de energías que van desde los neutrones térmicos hasta los neutrones del orden de los MeV. Sin embargo, la sensibilidad para cada esfera alcanza una energía neutrónica particular dependiendo del diámetro y la densidad de la esfera (Mazrou et al., 2016, Thomas and Alevra, 2002, Decker et al., 2015).

Las cuentas obtenidas de la combinación esfera detector del BSS conforman la matriz de respuesta. Por lo que las tasas de conteo y el espectro de neutrones están relacionados a través de la versión discreta de la ecuación integro-diferencial de Fredholm, que es un sistema mal condicionado con un número infinito de soluciones. Para una solución significativa del problema y así poder desplegar el espectro de neutrones se utilizan varios métodos, como algoritmos basados en inteligencia artificial, Montecarlo, parametrización y procedimientos iterativos (Mohammadi et al., 2015, Birattari et al., 2010, Vega-Carrillo et al., 2010).

Con el avance tecnológico de la inteligencia artificial, las Redes Neuronales Artificiales (ANNs) han demostrado ser un procedimiento alternativo en la espectrometría de neutrones, ya que han sido desarrolladas para emular el cerebro humano. Las ANNs *feedforward* multicapa están constituidas por tres capas, la capa de entrada que no tiene neuronas, una o varias ocultas y la capa de salida, que a su vez las capas están interconectadas en paralelo por medio de los pesos sinápticos donde se almacena el conocimiento. Para el entrenamiento se establece un conjunto definido de reglas para la solución de un problema por medio del algoritmo de aprendizaje, donde se aprende a través de un proceso iterativo de ajustes aplicados a sus pesos sinápticos y umbrales, una vez que se obtiene el mínimo error a la salida de la red los pesos sinápticos han quedado ajustados y la ANN esta lista para la simulación (Kardan et al., 2004, Ortiz-Rodríguez et al., 2014, Mohammadi et al., 2015, Vega-Carrillo et al., 2010).

Por consiguiente, el desarrollo de un buen modelo de una ANN depende de los datos utilizados para su entrenamiento, validación y pruebas, del algoritmo de aprendizaje empleado, así como de la arquitectura, complejidad y tamaño de la red (Hernandez-Davila et al., 2014).

Una Unidad Central de Procesamiento (CPU) tradicional presenta un buen rendimiento cuando se simula una red neuronal en pequeña escala, sin embargo para las ANNs con una arquitectura robusta y con una cantidad de datos grande, el principal inconveniente es el tiempo de ejecución que es extremadamente largo debido a que las ANNs son estructuras no lineales, por lo que la CPU se ve limitada al realizar cálculos computacionalmente intensivos. El gran número de operaciones y la relativamente baja transferencia de datos hace que las ANNs sean potencialmente atractivas para la computación en una Unidad de Procesamiento Gráfico (GPU) (Vincenzi et al., 2011, Naveros et al., 2013).

Una GPU es un circuito integrado dedicado al procesamiento de gráficos tridimensionales u operaciones de coma flotante, y su función es aligerar la carga de trabajo de la CPU para aplicaciones 3D interactivas. La GPU es un procesador de propósito específico, contiene múltiples núcleos de procesador más sencillos y eficientes, diseñados para la computación paralela, y se ha utilizado para implementar algoritmos computacionalmente intensivos en diversas áreas, incluyendo la computación de alto rendimiento, la computación científica, y el procesamiento de imágenes a través de su arquitectura masivamente paralela (Brette and Goodman, 2012, Oh and Jung, 2004, Tsuchida and Yoshioka, 2013, NVIDIA, 2017).

No es posible reemplazar la CPU por una GPU, sin embargo trabajando las dos unidades en conjunto, se puede multiplicar el rendimiento computacional en operaciones con vectores y matrices. No sólo es más eficiente, los semiconductores han reducido su tamaño y su precio. Por lo tanto, el uso eficiente de la GPU y su explotación han sido identificados como el próximo gran avance en la

automatización de diseño electrónico (Naveros et al., 2013, Ho et al., 2008, Sridhar et al., 2012, Vincenzi et al., 2011, NVIDIA, 2017).

Desde entonces, las GPUs se han instalado en centros de datos energéticamente eficientes de laboratorios gubernamentales, universidades, grandes compañías y PYMEs de todo el mundo. Las GPUs desempeñan un papel fundamental en la aceleración de aplicaciones que abarcan desde la biología computacional hasta los robots. Tras esto, con el fin de facilitar la computación de propósito general en una GPU, NVIDIA® ha proporcionado un entorno exclusivo de desarrollo integrado sin costo denominado CUDA (Compute Unified Device Architecture), que es una extensión del lenguaje de programación C y ofrece un conjunto de herramientas de desarrollo que permite a los programadores usar una variación del lenguaje, además de ser compatible con el software MATLAB®, que permite acelerar las aplicaciones con GPU de forma más fácil que mediante C o Fortran y sin tener que conocer los entresijos de las arquitecturas de las GPUs o las bibliotecas de cálculo GPU de bajo nivel. Ya que las aplicaciones de MATLAB se desarrollan en un lenguaje de programación propio (Brette and Goodman, 2012, Mei et al., 2014, Tsuchida and Yoshioka, 2013, NVIDIA, 2017, MATLAB, 2016e).

La computación gráfica ha alcanzado una fase en la que nuevas tecnologías y capacidades de modelado compatibles con Unidades de Procesamiento Gráfico de Propósito General (GPGPUs) pueden soportar el análisis de las ANNs, aunque pueden ser simuladas utilizando software tradicional, muchas aplicaciones potenciales requieren procesamiento en tiempo real. Por lo tanto las GPGPU hacen potencialmente posible para este trabajo determinar el espectro de neutrones mediante una ANN óptima implementada en GPGPU. Además es interesante comparar y establecer un vínculo definitivo entre las arquitecturas GPGPU, CPU, modelos de programación y las aplicaciones que se pueden acelerar con esta tecnología (Mei et al., 2014, Pallipuram et al., 2012, Vincenzi et al., 2011).

Revisando que no hay una implementación que involucre el BSS, ANN y GPGPU se plantea el siguiente problema científico.

### **Problema Científico**

Es difícil estructurar algoritmos para aprovechar los procesadores masivamente paralelos. En este contexto, las ANN pueden representar una solución óptima para desplegar el espectro de neutrones a partir de la combinación de las tasas de conteo obtenidas del BSS, donde se intenta aprovechar la potencia, arquitectura y tiempo de cálculo de las CPU, y GPU NVIDIA®, mediante el software MATLAB®, que permite acelerar aplicaciones no relacionadas con los gráficos en GPU.

### **Objetivo general**

Determinar el espectro de neutrones a partir de las 7 tasas de conteo obtenidas del BSS, con una ANN óptima, implementada en CPU y GPGPU con diferente número de núcleos CUDA.

### **Objetivos específicos**

1. Obtener las tasas de conteo de diferentes fuentes de neutrones con el BSS
2. Diseñar una ANN óptima
3. Entrenar la ANN con el algoritmo *trainscg*
4. Evaluar el funcionamiento de la ANN en CPU y GPU
5. Validar y probar la ANN con los datos obtenidos del BSS e IAEA

## II REVISIÓN DE LITERATURA

### 2.1 Neutrones

El primer avance sobre el descubrimiento del neutrón fue hecho en 1920 por Rutherford el cual estableció que el neutrón estaba formado por la unión de un protón y un electrón, también postulo una partícula "neutral" para tratar con las cuestiones de masa y carga, pero no encontró evidencia. La inclinación por disparar partículas alfa en sustancias no sólo descubrió la transmutación de elementos a través de interacciones nucleares, sino que sus descubrimientos condujeron a otros investigadores a hacer experimentos similares. Irène Curie y Frédéric Joliot, habían irradiado todos los elementos ligeros con partículas alfa y descubrieron que podían producir sustancias radiactivas artificialmente; Sin embargo, sus resultados para el berilio eran desconcertantes (Arya, 1966, Martín del Brío and Sanz Molina, 2006).

El descubrimiento del neutrón fue hecho en hasta 1932, por el físico británico J. Chadwick cuando repitió los experimentos de Joliet-Curie pero con la sospecha de que tal vez el producto era la partícula neutra de Rutherford. Los Joilot-Curie fueron de los primeros en producir y "observar" Neutrones, pero no se percataron de este hecho debido a que no sabían lo que eran. El descubrimiento resolvió el rompecabezas de la masa de átomos individuales y explicó la existencia de isótopos descubiertos en 1913 por Soddy para elementos radiactivos (Cember and Johnson, 2009, Magill and Galy, 2004, Arya, 1966).

El neutrón o una partícula neutra (hadrón) que es estable dentro del núcleo atómico y se vuelve inestable, cuando está fuera de éste, pero se desintegra en un protón ( $p$ ), un electrón ( $\beta^-$ ) y un antineutrino ( $\bar{\nu}_e$ ) con una vida media de 12 minutos fuera del núcleo. Su masa en reposo es  $1.6749286 \times 10^{-27}$  kg y es ligeramente mayor a la

del protón. Los neutrones se producen en todos los núcleos atómicos, excepto en el hidrógeno normal y son eléctricamente neutros. Su presencia en el núcleo explica la diferencia entre el número atómico y el número de masa atómica, también suministra la fuerza de cohesión que mantiene unido el núcleo. Esta fuerza se llama fuerza nuclear. Se cree que actúa sobre un rango extremadamente corto alrededor de  $2 \times 10^{-15}$  a  $3 \times 10^{-15}$  m (Cember and Johnson, 2009, Magill and Galy, 2004).

Por analogía con el caso ordinario de partículas cargadas, se puede suponer que el neutrón y el protón llevan ciertas cargas nucleares y que los campos de fuerza debido a estas cargas nucleares se establecen alrededor de los nucleones (partículas dentro del núcleo). Estudios recientes en la física de partículas han demostrado que los protones y los neutrones no son las partículas elementales. Se componen de conjuntos de tres partículas, llamadas quarks, que ahora se cree que son las partículas elementales. Los quarks son partículas cargadas, los que tienen  $\pm 2/3$  de la carga protónica se llaman quarks arriba y los que llevan  $\pm 1/3$  de la carga protónica se llaman quarks abajo. Un neutrón está hecho de un quark positivo hacia arriba y dos quarks abajo negativos, lo que resulta en una carga neta de cero (Cember and Johnson, 2009).

Los neutrones se clasifican en función de su energía porque el tipo de reacción que sufren depende en gran medida de esta. Los neutrones de alta energía, son aquellos cuyas energías superan unos 0.1 MeV, y se denominan *neutrones rápidos*. Los *neutrones térmicos*, por otra parte, tienen la misma distribución de la energía cinética que las moléculas de gas. Al respecto, los neutrones térmicos son indistinguibles de las moléculas de gas a la misma temperatura. Aunque los neutrones nacen rápido, también se someten rápidamente a diversas interacciones con los medios de comunicación y su energía se degrada. Los neutrones son mejor clasificadas en términos de sus energías mostrados en la Tabla 1. Las clases lentas e intermedias se usan indistintamente (Martin, 2006, Cember and Johnson, 2009).

Tabla 1. Clasificación de los neutrones por sus energías.

<b>Tipo de neutrones</b>		<b>Rango de energías</b>
<b>Fríos</b>	T < 20°C	<0.0253 eV
<b>Térmicos</b>		0.0253 eV
<b>Epitérmicos</b>	27°C, se extiende hasta el punto de corte de absorción fuerte por un absorbedor de Cd.	0.0253 a ~ 1 eV
<b>Epicadmios</b>	Captura de resonancia de 20000 barns a 0.4 eV, transparentes a un absorbedor de Cd.	> 1 eV
<b>Lentos</b>		0.0253 a 100 eV
<b>De energía intermedia</b>		0.5 a 10E(4) eV
<b>Rápidos</b>		0.01 a 10 MeV
<b>De alta energía</b>		>10 MeV

## Interacción de neutrones con la materia

Todos los neutrones desde su nacimiento son rápidos. Generalmente, los neutrones rápidos pierden energía colisionando elásticamente con los átomos en su entorno, y luego, después de ser ralentizados a energías térmicas, son capturados por los núcleos del material absorbente. Una interacción muy débil entre ellos se debe a sus momentos magnéticos. Dado que los neutrones son eléctricamente neutros, no tienen que cruzar la barrera de Coulomb, como resultado, la probabilidad (sección transversal), para las interacciones nucleares es mayor para los neutrones que para las partículas cargadas, y por tanto, incluso los neutrones muy lentos pueden alcanzar el núcleo sin ninguna dificultad. Cuanto más lento es el neutrón, más

probable es su captura por el núcleo. Aunque existen varios tipos posibles de reacciones de neutrones, las principales se muestran en la Tabla 2, donde la reacción de neutrones más común es la captura radiactiva (Arya, 1966, Magill and Galy, 2004, Martín del Brío and Sanz Molina, 2006, Lamarsh and Baratta, 2014, Tsoufanidis and Landsberger, 2011).

Tabla 2. Principales neutrones inducidos por reacciones nucleares.

Reacciones	Descripción
<b>Dispersión de neutrones</b> ( $n, n'$ )	Elástica: el neutrón ha sido dispersado por el núcleo, se conserva la energía cinética y el momento.
	Inelástica: parte de la energía cinética del neutrón incidente se utiliza en la excitación del núcleo del blanco y la energía de excitación se emite como un rayo gamma (fotón).
<b>Captura neutrónica</b> ( $n, \gamma$ )	Reacción nuclear exotérmica, un neutrón libre colisiona con un núcleo atómico, el neutrón es capturado y emite rayos gamma sin producir fisión.
<b>Emisión de partículas</b> ( $n, 2n$ ), ( $n, p$ ), ( $n, \alpha$ ) y ( $n, \gamma$ )	Se generan en reacciones nucleares o desintegración radiactiva de otros núcleos que transmutan en elementos más ligeros mediante la emisión de dichas partículas.
<b>Fisión</b>	Es la principal fuente de energía nuclear, ocurre cuando un núcleo pesado se divide en dos o más núcleos pequeños, neutrones libres, fotones y otros fragmentos del núcleo.

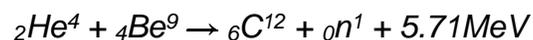
## Producción de neutrones

Dependiendo de la energía y la intensidad del haz de neutrones, se utilizan diferentes métodos para la producción de neutrones. La fuente más prolífica de

neutrones es el reactor nuclear. La división de uranio o de un núcleo de plutonio en un reactor nuclear se acompaña de la emisión de varios neutrones. Estos neutrones de fisión tienen una amplia gama de energías, la distribución alcanza un máximo de 0.7 MeV y tiene un valor medio de 2 MeV. Excepto por varios fragmentos de fisión de vida media muy corta, no existen radionucleidos que se desintegren al emitir neutrones. El Californio 252, sin embargo, experimenta una fisión nuclear espontánea a una tasa promedio de 10 fusiones por cada 313 transformaciones alfa y una vida media de 2.73 años. Todas las demás fuentes de neutrones dependen de reacciones nucleares para la emisión de neutrones y pueden producirse haces de neutrones copiosos en aceleradores por muchas reacciones diferentes (Cember and Johnson, 2009, Arya, 1966).

### Neutrones a partir de reacciones ( $\alpha, n$ )

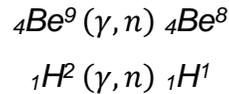
Los neutrones son producidos por el bombardeo de un elemento ligero por las partículas alfa. La reacción más común es aquella en la que el berilio se utiliza como blanco. Otros elementos ligeros utilizados son Li, F y B. Las partículas alfa se pueden obtener de polonio, radón, o radio que es la fuente más útil de los neutrones, las partículas alfa son absorbidas en distancias muy cortas en los sólidos, por lo que es necesario hacer una mezcla profunda de radio y el berilio (Arya, 1966).



### Neutrones a partir de fotodesintegración ( $\gamma, n$ )

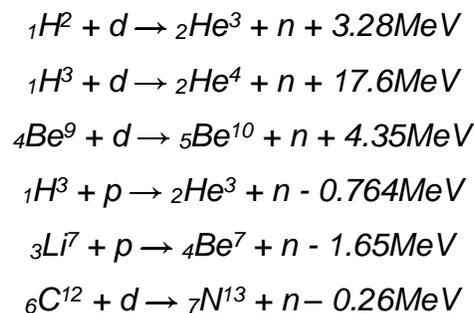
Es posible obtener un haz de rayos gamma intenso a partir de núcleos radiactivos artificiales producidos en reactores nucleares. Estos a su vez pueden ser utilizados

para producir haces de neutrones de alto flujo por medio de fotodesintegración. Los neutrones producidos por fotodesintegración son más monoenergéticos que los de reacciones ( $\alpha, n$ ), y las energías de umbral para las siguientes dos reacciones son 2.23 MeV y 1.67 MeV, respectivamente (Arya, 1966).



### Neutrones a partir de aceleradores de partículas

Para la producción de neutrones monoenergéticos, el método más comúnmente utilizado es el bombardeo de ciertos blancos por las partículas cargadas aceleradas. Las reacciones generalmente empleadas son las siguientes, de las cuales las dos primeras reacciones se utilizan con más frecuencia (Arya, 1966, Martin, 2006, Magill and Galy, 2004, Tsoufanidis and Landsberger, 2011).



### Neutrones a partir de reacciones de extracción

Los neutrones de muy alta energía pueden ser producidos por medio de extracción de deuterones. El deuterón se compone de un neutrón y un protón unidos por una

energía de sólo 2.2 MeV. Cuando un haz de deuterones de muy alta energía golpea un blanco, es posible que el neutrón y el protón puedan ser separados. Esto da lugar a la emisión del neutrón con aproximadamente la mitad de la energía de deuterones, mientras que el protón puede ser capturado por el núcleo (Arya, 1966).

### Neutrones a partir de protones de alta energía

Es uno de los medios más simples de producción de neutrones de alta energía por interacción nucleón-nucleón. Utiliza choque frontal entre un protón incidente de muy alta energía y un neutrón como blanco. Los neutrones eliminados en la dirección de avance llevan toda la energía y el impulso del protón incidente (Arya, 1966).

### Neutrones a partir de reactores de fisión nuclear

Para la producción de haces de neutrones de baja energía de altos flujos, los reactores nucleares no pueden ser superados por ningún otro método. Las energías de neutrones emitidos por la fisión varían de energías térmicas a aproximadamente 18 MeV, con un valor promedio de energía de neutrones de 2 MeV. Los neutrones en los reactores nucleares son producidos por el proceso de fisión. En el proceso de la fisión, un neutrón es absorbido por un núcleo fisionable, y se rompe el núcleo en dos partes casi iguales con una emisión de más de un neutrón para cada neutrón absorbido (Arya, 1966).

## 2.2 Detección de neutrones

La detección de neutrones como de los rayos gamma es un problema mucho más complejo que el de detección de partículas cargadas y fotones, debido a que los neutrones no llevan ninguna carga eléctrica y no causan ionización directa. Deben reaccionar con otro medio para producir una partícula ionizante primaria. La mayoría de los métodos que se discutirán dependen directa o indirectamente en los siguientes principios: (a) La interacción de un neutrón con un núcleo, que incluye la dispersión elástica, emisión de partículas cargadas en una reacción nuclear, causado por los neutrones y la producción de un núcleo radiactivo, (b) el método de tiempo de vuelo, y (c) la difracción de neutrones. Debido a la fuerte dependencia de la velocidad de reacción de neutrones en la sección transversal para esa reacción particular, se utilizan medios de detección diferentes, dependiendo de la energía de los neutrones que se está tratando de medir. Cabe señalar que en la mayoría de los casos la energía de los neutrones está predeterminado por el método de su producción (Cember and Johnson, 2009, Arya, 1966).

### Sistema espectrométrico de esferas Bonner (BSS)

El sistema más utilizado para medición de neutrones relacionado con la protección radiológica es el espectrómetro de esferas Bonner (BSS). El BSS se introdujo por primera vez en 1960 por R. L. Bramblett, R. I. Ewing y T. W. Bonner. Un sistema típico consiste en una serie de esferas de polietileno de alta densidad de ( $0.95 \text{ g/cm}^3$ ), cuyos diámetros van desde 5.08 cm a 30.48 cm (2 a 12 pulgadas) son: 0" (detector desnudo), 2", 3", 5", 8", 10" y 12" y sirven como material moderador, y un detector de neutrones térmicos que se coloca en forma alternada en el centro de las esferas de polietileno. El BSS se calibró como una función de la energía de neutrones de 0.05 a 15.1 MeV utilizando neutrones monoenergéticas de reacciones

entre partículas cargadas (Bramblett et al., 1960, Howell et al., 2010, Burgett et al., 2009, Cruzate et al., 2009). En la Figura 1, se muestran los moderadores del BSS.

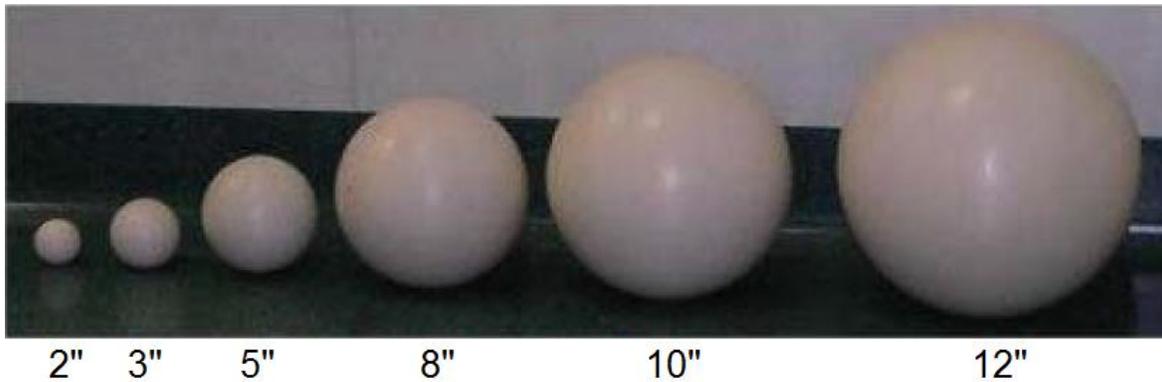


Figura 1. Conjunto de moderadores del BSS (Cruzate et al., 2009).

El BSS es un espectrómetro de neutrones ampliamente utilizado porque responde a un amplio intervalo de energía, alrededor de los 15 MeV y tiene una respuesta casi isotrópica. Sin embargo, la sensibilidad del BSS disminuye drásticamente. Se ha investigado el uso de capas concéntricas de cobre (Cu), tungsteno (W), y plomo (Pb) de alto número atómico (Z), y así aumentar la sensibilidad del sistema para altas energías. Un esquema general del BSS con el detector de neutrones de  $^3\text{He}$  se muestra en la Figura 2 (Thomas and Alevra, 2002, Howell et al., 2010, Burgett et al., 2009, Vega-Carrillo, 2002, Bramblett et al., 1960, Cruzate et al., 2009).

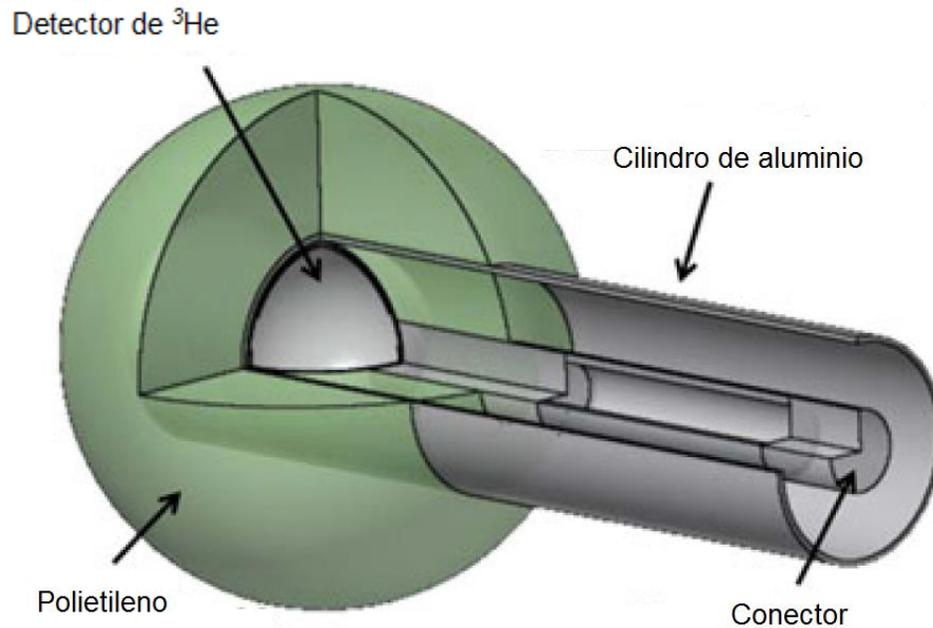
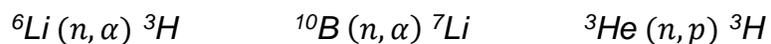


Figura 2. Esquema de contador de neutrones (Aza et al., 2016).

El BSS también usa otro tipo de detectores de neutrones térmicos, en su diseño original el detector era un centellador de  $^6\text{LiI}(\text{Eu})$  (Ioduro de litio dopado con europio) de 4x4 mm de longitud. También se han usado los detectores gaseosos de  $\text{BF}_3$  (trifloruro de boro) o detector de  $^3\text{He}$ .



En el caso del  $^6\text{LiI}(\text{Eu})$  contiene 96.1% de isótopo  $^6\text{Li}$  que en forma natural solo existe con una abundancia del 7.5%. Alrededor del 80% de los neutrones térmicos incidentes son absorbidos en 1.0 mm de  $^6\text{LiI}(\text{Eu})$ , por lo que la detección de neutrones térmicos es esencialmente un efecto de superficie. Este espectrómetro tiene la ventaja del uso de circuitos electrónicos simples, los flujos de neutrones pueden ser detectados con presencia de gammas y la sensibilidad de estos detectores es moderadamente alta para los neutrones (Khabaz and Miri, 2011,

Bramblett et al., 1960, Mares and Schraube, 1994, Tsoufanidis and Landsberger, 2011).

El contador tiene una excelente sensibilidad de energía de 0.1 a 2 MeV y es particularmente útil para la determinación de las formas de los espectros de neutrones continuos. A medida que el diámetro de la esfera se incrementa, la respuesta del espectrómetro se desplaza a energías de neutrones más altas. En consecuencia, en las esferas de mayor tamaño, los neutrones deben atravesar un camino más largo después de termalización, y son más propensos a ser capturado antes de alcanzar el cristal  ${}^6\text{LiI}(\text{Eu})$ . Cada esfera de polietileno con el centelleador tiene una respuesta única a los neutrones monoenergéticos, el conjunto de las respuestas individuales se conoce como matriz de respuesta de esfera Bonner (Bramblett et al., 1960, Vega-Carrillo, 2002, Mares and Schraube, 1994, Arya, 1966, Cruzate et al., 2009, Tsoufanidis and Landsberger, 2011).

## Medición con el BSS

Los neutrones que sufren eventos de dispersión dentro de un moderador pierden energía hasta que son absorbidos o escapan al moderador. El polietileno se utiliza en las esferas del moderador del BSS, debido a la gran proporción de sus secciones transversales de dispersión a absorción. Tal relación maximiza el número posible de neutrones incidentes que pueden llegar a ser termalizados antes de alcanzar el cristal  ${}^6\text{LiI}(\text{Eu})$ , como se demuestra en la Figura 3.

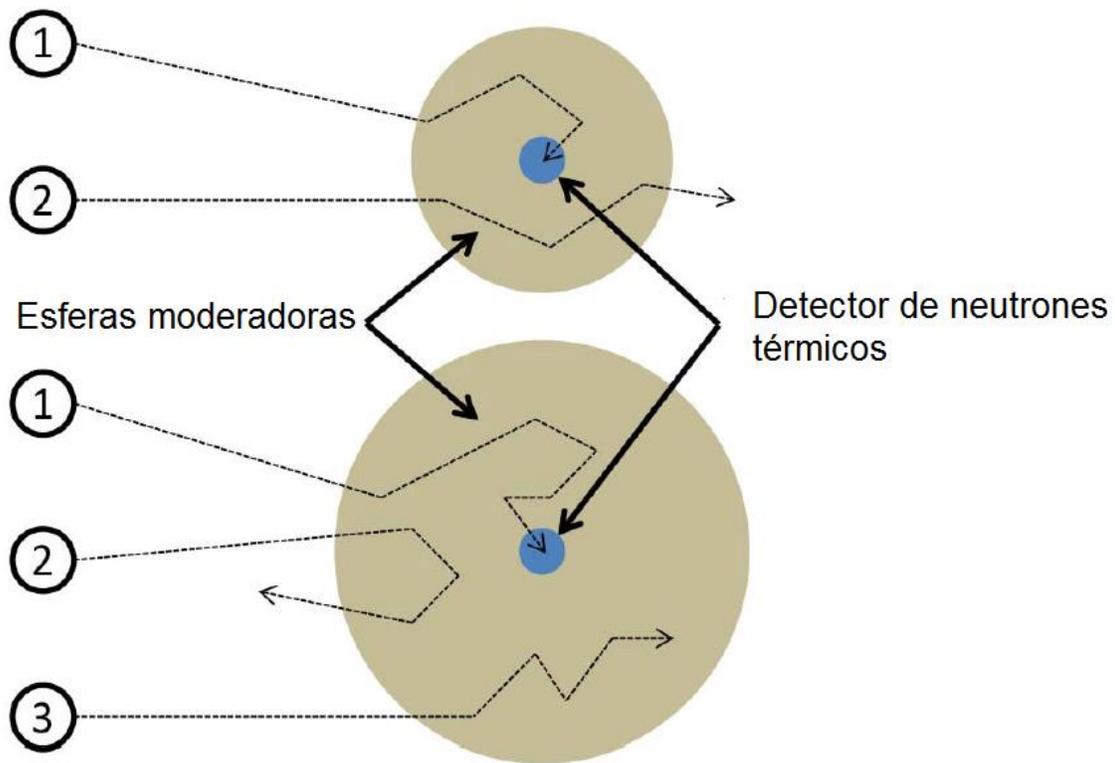


Figura 3. Esquema de posibles pistas de neutrones en dos esferas Bonner de diferentes tamaños (Decker et al., 2015, Knoll, 2000).

La pista 1 representa neutrones moderados por el polietileno y detectados por el cristal, la pista 2 representa neutrones moderados parcialmente que se escapan, y la pista 3 muestra cómo los neutrones pueden sufrir captura parásita dentro del material moderador. Las esferas más grandes tienden a mejorar la frecuencia de la pista 3, mientras que reduce la frecuencia de la pista 2. El detector de neutrones térmicos se inserta en una esfera a la vez y los neutrones se cuentan durante un intervalo de tiempo determinado, normalmente de 5 a 30 minutos. Esto continúa hasta que los recuentos se adquieren con cada esfera y con el detector desnudo. El número de recuentos obtenidos con el contador de  ${}^6\text{Li}(\text{Eu})$  en cada esfera depende de la energía incidente y de la función de respuesta de la esfera. El

comportamiento general observado en este tipo de estudio se muestra en la Figura 4. La diferencia en las formas y posición de los máximos en estas curvas de respuesta sirve como base para usar el conjunto de esferas como un espectrómetro de neutrones simple. Esfera resulta tener una forma similar comparada con el equivalente de dosis suministrado por neutrón en función de la energía (Dubeau et al., 2011, Decker et al., 2015, Knoll, 2000).

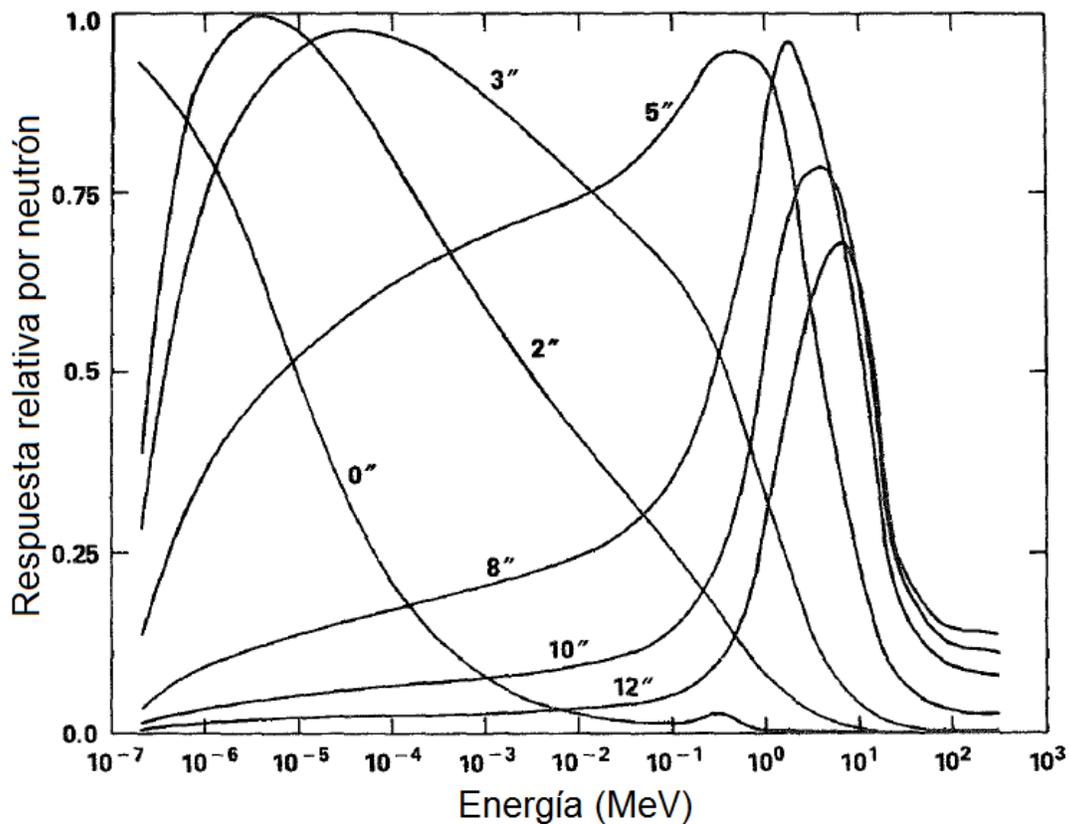


Figura 4. Dependencia energética de las eficiencias de detección relativas del BSS (Knoll, 2000).

De lo anterior las funciones de respuesta, incluso de los grandes moderadores llegan cerca del pico de 10 MeV y rápidamente caen a energías más altas. Además, en los espectros de energías superiores a los 20 MeV y en esferas más grandes, la

señal no se puede distinguir. Como resultado, estos sistemas no son fiables para reconstruir los espectros de neutrones para aceleradores de alta energía. La manipulación de las esferas de más de 12 pulgadas son difíciles ya que agravan más las dificultades anteriores, en la Tabla 3 se muestran las constantes de proporcionalidad calculadas para la modelación de funciones de respuesta del BSS en MCNP6 (Vylet, 2002, Duke et al., 2016).

Tabla 3. Constantes de proporcionalidad calculadas.

<b>Detector</b>	<b>Constante de proporcionalidad</b>
<b>0 pulgadas (detector desnudo)</b>	$1.400 \times 10^{-04}$
<b>2 pulgadas</b>	$1.774 \times 10^{-02}$
<b>3 pulgadas</b>	$3.990 \times 10^{-02}$
<b>5 pulgadas</b>	$1.109 \times 10^{-01}$
<b>8 pulgadas</b>	$2.838 \times 10^{-01}$
<b>10 pulgadas</b>	$4.435 \times 10^{-01}$
<b>12 pulgadas</b>	$6.386 \times 10^{-01}$

La principal ventaja de la BSS es su alta sensibilidad a todo el rango de energía de neutrones de interés. Típicamente, los máximos en la respuesta corresponde de 1 a 4 conteos del detector para la unidad de fluencia (recuentos\*cm<sup>2</sup>). Por lo tanto, los datos de muy alta calidad estadística se pueden adquirir en pocos minutos, por el contrario, las desventajas son la necesidad de tiempos de exposición relativamente largos debido a la irradiación secuencial de las esferas, y la resolución de energía limitada, especialmente en la región epitérmica. Otra desventaja del sistema es su peso, más de 16 kg para las esferas más la electrónica, y su volumen. Además, el manejo real de las esferas y el detector en un entorno operativo es tedioso (Braga and Dias, 2002, Dubeau et al., 2011, Aza et al., 2016).

## Matriz de respuesta del detector

La salida del BSS no proporciona directamente los espectros de neutrones. Al momento que los neutrones de cierta energía atraviesan una esfera, pierden energía y alcanzan al detector produciendo una señal que se traduce en una tasa de conteo. La salida no es más que un recuento del número de reacciones  $(n, \alpha)$  en el centellador durante un período de tiempo de exposición determinado o actividad total obtenida durante un período de tiempo de exposición dado. En la ecuación 1, se muestra la relación entre las tasas de conteo en función del diámetro de la esfera  $C(D)$ , la matriz de respuesta  $R(D, E)$  y el espectro de los neutrones  $\varphi_E(E)$ . Ésta se conoce como ecuación integro-diferencial de Fredholm de primer tipo:

$$C(D) = \int_{E_{min}}^{E_{max}} R(D, E)\varphi(E_n)dE_n \dots\dots\dots 1$$

En la ecuación 2, se muestra la versión discreta de la ecuación anterior, donde  $C_j$  es la tasa de conteo del  $j$  – ésimo detector  $R_{i,j}$  es la matriz de respuesta del  $j$  – ésimo detector de neutrones en el  $i$  – ésimo intervalo de energía,  $\varphi_i$  es la fluencia de neutrones en el  $i$  – ésimo intervalo de energía y  $m$  es el número de esferas del sistema espectrométrico (Avdic et al., 2006, Sweezy et al., 2002).

$$C_j = \sum_{i=1}^m R_{i,j} \varphi_i, m = 1, 2, 3, \dots, \text{número de esfera} \dots\dots\dots 2$$

Es deseable producir un espectro de salida con muchos grupos de energía. Sin embargo, el sistema de ecuaciones, es decir, el número de grupos de energía desconocidas es mayor al número de ecuaciones (arreglos de detectores), de tal

forma que la ecuación de la matriz está indeterminada, no hay una solución única para esta ecuación, se le llama sistema de ecuaciones mal condicionado, para lo cual se implementa el uso de las ANN. Esto es especialmente cierto cuando se trata de desplegar en las regiones de energía más altas, donde hay puntos de datos únicos limitadas en los datos de respuesta del BSS (Sweezy et al., 2002, Burgett et al., 2009).

### 2.3 Redes neuronales artificiales

El primer paso hacia las ANN se dio en 1943, cuando entre el neurobiólogo, Warren McCulloch y estadístico Walter Pitts, publicaron un artículo seminal titulado "*Un cálculo lógico de ideas inminentes en la actividad nerviosa*", en el Boletín de Matemática Biofísica. Es un modelo electrónico simple que puede ser implementado por hardware. El artículo inspiró el desarrollo de la computadora digital moderna, o el cerebro electrónico, como lo llamó John von Neumann. Aproximadamente al mismo tiempo, Frank Rosenblatt también fue motivado por este trabajo para investigar la computación del ojo, que finalmente llevó a la primera generación de redes neuronales, conocido como el perceptrón. La investigación sobre la ANN se ha llevado a cabo a partir de la observación de cómo es que el cerebro humano realiza tareas de reconocimiento, estas metodologías resultan muy distintas a las que efectúa una computadora digital (Haykin, 2007, Martín del Brío and Sanz Molina, 2006, Hu and Hwang, 2002, Chow and Cho, 2007).

Una ANN es un paradigma computacional matemático general, está inspirada en los sistemas nerviosos biológicos. Las redes neuronales se componen de elementos simples que funcionan en paralelo. Una ANN modela las operaciones de los sistemas neuronales biológicos, que sirve para resolver problemas de forma individual o combinadas con otros métodos, para aquellas tareas de clasificación,

identificación, diagnóstico, optimización o predicción en las que el balance datos/conocimiento se inclina hacia los datos y donde, adicionalmente, puede haber la necesidad de aprendizaje en tiempo de ejecución y de cierta tolerancia a fallos (Mohammadi et al., 2015, Hu and Hwang, 2002, Tablada and Torres, 2009).

Las características principales de las ANN son las siguientes (Haykin, 2007, Hu and Hwang, 2002, Hernandez-Davila et al., 2014, Basheer and Hajmeer, 2000, Martín del Brío and Sanz Molina, 2006):

- Aprendizaje adaptativo: Pueden comportarse en función de un entrenamiento con una serie de ejemplos ilustrativos. De esta forma, no es necesario elaborar un modelo a priori, ni establecer funciones probabilísticas. Una ANN es adaptativa, porque puede modificarse constantemente con el propósito de adaptarse a nuevas condiciones de trabajo.
- Auto organización: Consiste en la modificación de toda la red completa con el fin de llevar a cabo un objetivo específico. Auto organización significa generalización, de esta forma, una red puede responder a datos o situaciones que no ha experimentado antes, pero que puede inferir en base a su entrenamiento. Esta característica es muy útil, sobre todo cuando la información de entrada, es poco clara o se encuentra incompleta.
- Tolerancia a fallos: La tolerancia a fallos se entiende en dos sentidos: primero, las redes pueden reconocer patrones de información con ruido, distorsión o incompletos (tolerancia de fallos respecto de los datos); y segundo, pueden seguir trabajando con cierta degradación aunque se destruya parte de la red (tolerancia a fallos respecto de la estructura). La explicación de este fenómeno, se encuentra en que mientras la computación tradicional almacena la información en espacios únicos, localizados y direccionables, las ANN lo hacen de forma distribuida y con un alto grado de redundancia.

- Operación en tiempo real: de todos los métodos existentes, la ANN son las más indicadas para el reconocimiento de patrones en tiempo real, debido a que trabajan en paralelo, al actualizar simultáneamente todas sus instancias. Es importante destacar que esta característica, solo se aprecia cuando se implementan redes con hardware especialmente diseñado para el procesamiento en paralelo.
- Fácil inserción en la tecnología existente: es relativamente sencillo obtener chips especializados para redes neuronales que mejoran su capacidad en ciertas tareas así como su bajo consumo de energía. Ello facilita la integración modular en los sistemas existentes.

## Modelo neuronal biológico

La neurona es la unidad celular fundamental de un sistema nervioso y el cerebro humano corresponde al de un sistema altamente complejo, no lineal y paralelo. Lo anterior, en términos sencillos equivale a decir que puede realizar muchas operaciones simultáneamente a diferencia de los computadores comunes que son de tipo secuencial, o sea, realizan sólo una operación a la vez (Martín del Brío and Sanz Molina, 2006, Izaurieta and Saavedra, 1999).

En la Figura 5, se muestra una típica neurona biológica del cerebro humano. La función principal de las neuronas es la transmisión de los impulsos nerviosos. Estos viajan por toda la neurona comenzando por las dendritas hasta llegar a las terminaciones del axón, que son los caminos de entrada de otras neuronas, a través de una unión llamada sinapsis. En el cuerpo de la neurona o soma, se suman las entradas de todas las dendritas. Si estas entradas sobrepasan un cierto umbral, entonces se transmitirá un pulso a lo largo del axón, en caso contrario, no se

transmitirá (Martín del Brío and Sanz Molina, 2006, Chow and Cho, 2007, Tablada and Torres, 2009).

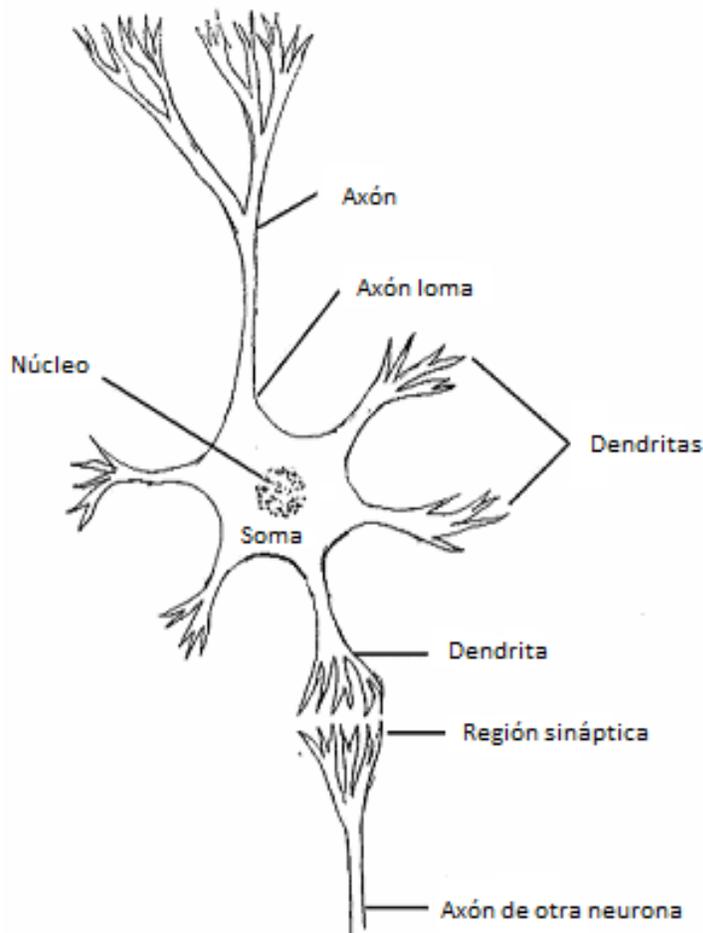


Figura 5. Modelo de una neurona biológica y su conexión (Chow and Cho, 2007).

Una neurona puede recibir desde unos cuantos miles hasta quince mil entradas de los axones de otras neuronas. Aparentemente esto forma un sistema paralelo masivo comparado con la arquitectura de computadora digital. En cada neurona, si la señal combinada excede un valor umbral, activa la activación de una neurona que produce una señal de salida a través del axón. La transmisión de información a través de sinapsis es de hecho un producto químico en la naturaleza, pero tiene

efectos secundarios eléctricos que podemos medir. Las actividades eléctricas en las neuronas aparecieron en una forma de pulso con una frecuencia en el rango de 1 KHz (Chow and Cho, 2007).

## Modelo matemático

El modelo de una neurona artificial es una imitación del proceso de una neurona biológica. Existen varias formas de nombrar una neurona artificial: nodo, neuronodo, celda, unidad o elemento de procesamiento. Un modelo neuronal simple es el elemento de procesamiento más fundamental de las redes neuronales. En la Figura 6, se muestra un modelo matemático de una neurona artificial (Hernandez-Davila et al., 2014).

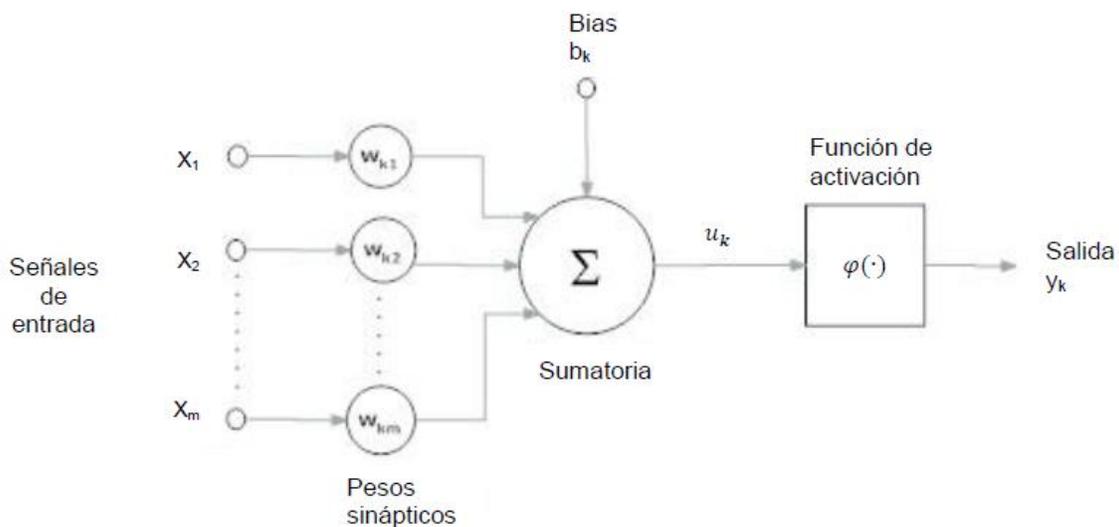


Figura 6. Modelo matemático de una neurona artificial (Hernandez-Davila et al., 2014).

Como se aprecia en la figura anterior, de la observación detallada del proceso biológico se han encontrado las siguientes analogías con el sistema artificial (Haykin, 2007, Martín del Brío and Sanz Molina, 2006):

- Las entradas  $X_1, X_2, \dots, X_n$ , representan las señales que provienen de otras neuronas y que son capturadas por las dendritas.
- Los pesos  $W_{k1}, W_{k2}, \dots, W_{km}$  son la intensidad de la sinapsis que conecta dos neuronas; tanto  $X$  como  $W$  son valores reales. El primer subíndice denota a la neurona hacia donde se dirige la información y el segundo subíndice, denota de qué neurona procede la información.
- Lo que realiza cada peso sináptico, es simplemente, multiplicar a su entrada correspondiente. Al respecto es necesario tener presente que en el soma de la neurona biológica, se suman las entradas provenientes de todas las dendritas, es por esto, que aquí aparece un sumador para sumar todas las señales de entrada multiplicadas por su peso sináptico. Esta suma se multiplica por una función de activación  $\varphi(\cdot)$ , para limitar la amplitud de la señal de salida de la neurona.
- La función de activación  $\varphi(\cdot)$  es seleccionada de acuerdo a la tarea realizada por la neurona. Algunos ejemplos de funciones de activación comunes se muestran en la Figura 7.
- Las señales de entrada a una neurona artificial  $X_1, X_2, \dots, X_n$ , son variables continuas en lugar de pulsos discretos. Cada señal de entrada pasa a través de una ganancia o peso, llamado peso sináptico o fortaleza de la conexión cuya función es análoga a la de la función sináptica de la neurona biológica.
- Los pesos pueden ser positivos (excitatorios), o negativos (inhibitorios), el nodo sumatorio acumula todas las señales de entrada multiplicadas por los pesos o ponderadas y las pasa a la salida a través de una función umbral o función de transferencia.

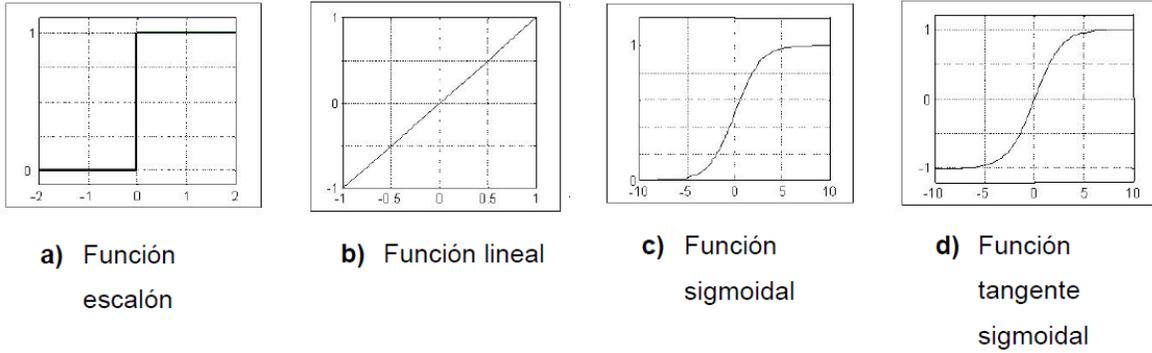


Figura 7. Diferentes tipos de funciones de activación (Hernandez-Davila et al., 2014).

La función *tangente sigmoïdal* es la más utilizada en la construcción de ANN, ya que está definida como una función estrictamente creciente que exhibe un buen balance entre un comportamiento lineal y uno no-lineal (Haykin, 2007).

En términos matemáticos, se puede describir a una neurona  $k$  con las ecuaciones 3 y 4, donde  $W_{kj}$  son los pesos sinápticos de la neurona, donde el subíndice  $k$  representa la neurona anterior y  $j$  la neurona siguiente;  $X_j$  son los valores de las señales de entrada;  $m$  es el número de neuronas de la capa de salida;  $u_k$  es el resultado de la sumatoria;  $b_k$  es el *bias*;  $\varphi(\cdot)$  es la función de activación y  $y_k$  es la señal de salida de la neurona (Haykin, 2007, Martín del Brío and Sanz Molina, 2006):

$$u_k = \sum_{j=1}^m W_{kj} X_j \dots\dots\dots 3$$

$$y_k = \varphi(u_k + b_k) \dots\dots\dots 4$$

El *bias* es un número cuyo valor no supera  $a + 1$  y tiene la función de mejorar las propiedades de convergencia de la red y ofrecer un nuevo efecto sobre el umbral de la función de activación que opera, ya sea incrementándolo o reduciéndolo, dependiendo de si el valor que toma es positivo o negativo, respectivamente (Haykin, 2007, Martín del Brío and Sanz Molina, 2006).

Una idea clara de este proceso se muestra en la Figura 8, en donde puede observarse el recorrido de un conjunto de señales que entran a la red.

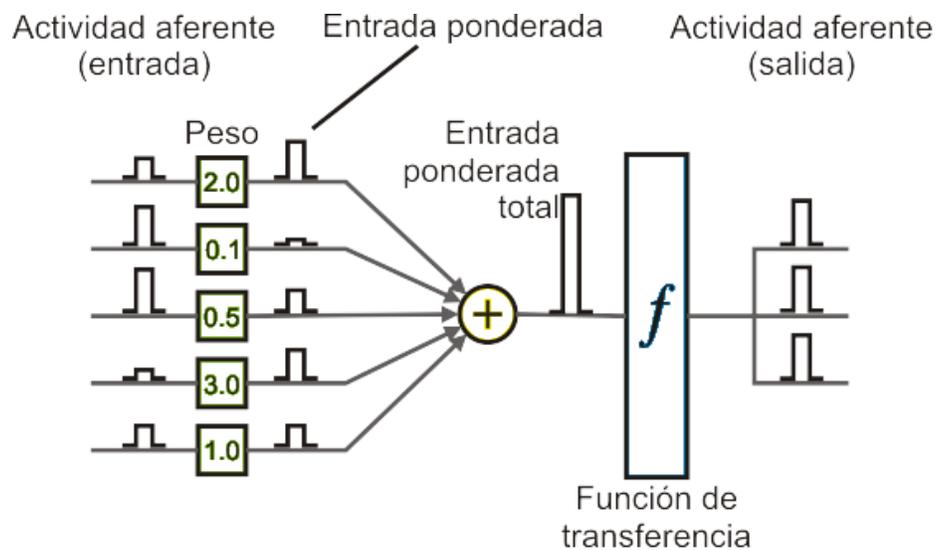


Figura 8. Operación de una neurona simple.

Dentro de una ANN, los elementos de procesamiento se encuentran agrupados por un mínimo de 3 capas, una capa es una colección de neuronas; de acuerdo a la ubicación de la capa en la red neuronal, ésta recibe diferentes nombres (Haykin, 2007):

- Capa de entrada. Recibe las señales de la entrada de la red, algunos autores no consideran el vector de entrada como una capa, pues allí no se lleva a cabo ningún proceso.
- Capas ocultas. Estas capas son aquellas que no tienen contacto con el medio exterior, sus elementos pueden tener diferentes conexiones y son éstas las que determinan las diferentes topologías de la red.
- Capa de salida. Recibe la información de la última capa oculta y transmite la respuesta al medio externo.

## Arquitectura

Una ANN, se basa en el concepto del funcionamiento del cerebro humano. En la mayoría de las topologías de red propuestas, el diseño de sus correspondientes algoritmos de entrenamiento se considera esencial. Una arquitectura de red exitosa debe ser soportada por un algoritmo de entrenamiento suficientemente eficaz. Las redes neuronales pueden clasificarse en dos tipos principales, a saber, las redes supervisadas y las redes no supervisadas. En términos generales, las redes neuronales supervisadas pueden subdividirse en tres clases fundamentales que son el modelo perceptrón, que es el tipo de red más simple, las redes recurrentes, que tienen uno o más bucles de realimentación y la red *feedforward*, siendo la más popular (Chow and Cho, 2007, Haykin, 2007).

### Red *Feedforward*

Una red *feedforward* puede ser unicapa y consiste en las entradas y una sola capa de neuronas. Tal red de alimentación de capa única no es capaz de clasificar

patrones separables, no linealmente. En las redes de *feedforward*, todas las conexiones son acíclicas o unidireccionales desde las entradas a la de salida. La red *feedforward* multicapa, consiste en una o más capas de neuronas, llamadas capas ocultas, entre las entradas y la capa de salida. Las neuronas en capas ocultas son llamadas neuronas ocultas, estas capas están conectados por pesos sinápticos en donde se almacena el conocimiento, como se muestra en la Figura 9. En las entradas de la ANN se reciben las características de entrada de una aplicación dada. Aunque la estructura de la red puede variar en cuanto al número de capas y número de nodos o neuronas, cada capa realiza la misma operación, la función de las neuronas ocultas, es la de intervenir de alguna manera útil entre las entradas y salidas de la ANN. Al agregar una o más capas ocultas, la red es capaz de manejar una tarea relativamente compleja y una clasificación lineal no separable (Haykin, 2007, Martín del Brío and Sanz Molina, 2006, Hu and Hwang, 2002, Oh and Jung, 2004, Chow and Cho, 2007).

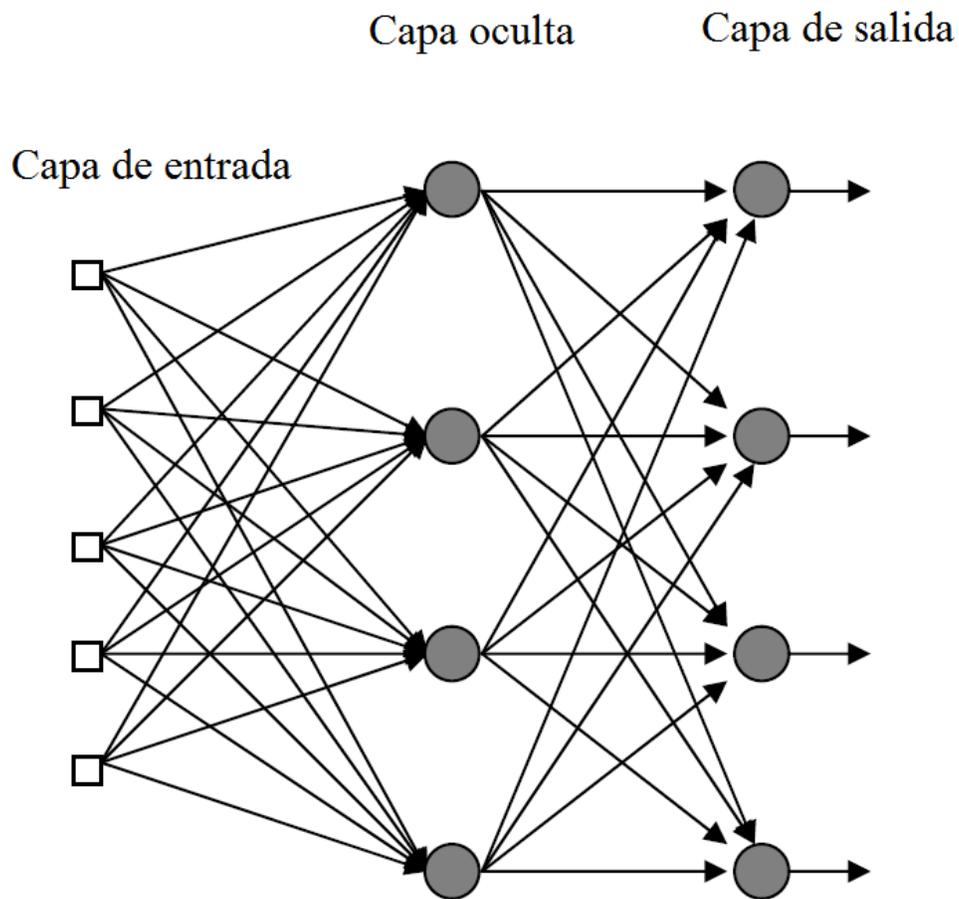


Figura 9. Red *feedforward* con una capa oculta (Chow and Cho, 2007).

### Proceso de aprendizaje

La propiedad más importante de una red neuronal, es la habilidad de la red de aprender de su medio ambiente y de mejorar su desempeño o respuesta, a partir del aprendizaje. Una red aprende de su medio a través de un proceso interactivo de ajustes sobre los pesos sinápticos y los niveles de umbral o *bias*. En la ANN, toda la información adquirida se guarda en el valor de cada peso sináptico. El

procedimiento de aprendizaje continúa hasta que la señal de error es cercana a cero o por debajo de un valor predefinido. Idealmente, la red adquiere más conocimiento después de cada iteración del proceso de aprendizaje hasta que se consigue el patrón entre los datos de entrada y salida (Haykin, 2007, Basheer and Hajmeer, 2000, Martín del Brío and Sanz Molina, 2006, Izaurieta and Saavedra, 1999).

El proceso de aprendizaje implica la siguiente secuencia de eventos (Haykin, 2007, Martín del Brío and Sanz Molina, 2006):

- La red neuronal es estimulada por un entorno.
- La red neuronal realiza cambios en sus parámetros libres como resultado de esta estimulación.
- La red neuronal responde en una nueva manera al entorno debido a los cambios que han ocurrido en su estructura interna.

El aprendizaje se divide principalmente en dos tipos: aprendizaje supervisado y aprendizaje no supervisado. La diferencia del aprendizaje supervisado radica en la inclusión de un profesor en su proceso de aprendizaje, donde la red necesita ejemplos de datos de formación consistentes en un número de pares de entrada-salida. En cambio en el aprendizaje no supervisado, se lleva a cabo a partir de un comportamiento auto-organizado. En el entrenamiento, no se utiliza ningún factor externo para afectar el ajuste de pesos de la red. Las salidas correctas no están disponibles durante el entrenamiento, esto es que no se cuentan con datos de salida para entrenar la red. En general, el rendimiento ANN depende de la calidad de los datos utilizados durante el entrenamiento y la arquitectura de red (Haykin, 2007, Martín del Brío and Sanz Molina, 2006, Chow and Cho, 2007, Mohammadi et al., 2015, Martínez-Blanco et al., 2016, Hernandez-Davila et al., 2014).

Un algoritmo de aprendizaje, es un conjunto de reglas bien definidas que están prescritas para la solución de un problema de aprendizaje. Básicamente, estos algoritmos de aprendizaje difieren de los demás, en la manera en la cual, el ajuste de los pesos sinápticos es formulado. Otro factor a considerar, es la manera en la cual una red neuronal, relaciona sus neuronas interconectadas con su medio (Haykin, 2007, Martín del Brío and Sanz Molina, 2006).

### Algoritmo de entrenamiento SCG

El algoritmo de entrenamiento de Gradiente Escalado Conjugado (SCG), fue desarrollado por Martin F. Moller y es un algoritmo de aprendizaje supervisado con tasa de convergencia superlineal. El algoritmo está basado en una clase de técnicas de optimización bien conocidas en el análisis numérico, como los métodos de gradiente conjugado. El SCG hace uso de información de segundo orden de la red neuronal, pero requiere solo una cantidad de uso de memoria en función del número de pesos presentes en la red neuronal. El aumento en velocidad depende del criterio de convergencia, ante mayor demanda de reducción del error, mayor será el aumento de velocidad. Además, de ser totalmente automatizado, evita la búsqueda de línea que provoca gran consumo de tiempo, la cual utilizan algoritmos de gradiente conjugado de *backpropagation* (retro-propagación) para cada iteración, con el fin de determinar el tamaño de paso apropiado (Møller, 1993).

Sin embargo la computación de ANN con muchas neuronas es un gran desafío, y es sujeto de considerable esfuerzo. La arquitectura de Von Neumann es una forma muy organizada de procesar la computación. Se logra utilizando una CPU que opera una serie de instrucciones que incluyen: buscar, decodificar, ejecutar y recuperar. La llegada de tarjetas de procesamiento de gráficos asequibles adecuadas para la computación paralela, se ha identificado como una tecnología para acelerar

numerosos algoritmos de datos en paralelo y por lo tanto hacen potencialmente posible escalar modelos de ANN más cerca de los tamaños de redes de estructuras cerebrales. Por ello las ANNs pueden ser reemplazadas ya que son más apropiadas para la aplicación en GPU, que se basa en el modelo circulante (Oh and Jung, 2004, Fidjeland et al., 2013, Slazynski and Bohte, 2012).

## 2.4 GPU

La GPU se desarrolló originalmente para la representación gráfica que requiere computación paralela con intensas operaciones aritméticas. Mientras que una GPU es un coprocesador de propósito específico, que consta de cientos o miles de núcleos computacionales más pequeños y eficientes, diseñados para la computación paralela. La CPU es un procesador de propósito general y realiza el procesamiento en serie, normalmente consta de cuatro a ocho núcleos de CPU. Por lo tanto, en los diseños de GPU, los transistores están dedicados al procesamiento de datos en lugar de almacenamiento de datos y control de flujo. Una GPU es un conjunto de multiprocesadores de una sola instrucción para múltiples datos SM o SIMD por sus siglas en inglés Single Instruction Multiple Data (Slazynski and Bohte, 2012, Mei et al., 2014, Thibault and Senocak, 2012, NVIDIA, 2016).

Las GPU modernas proporcionan actuaciones de ancho de banda, de memoria y de coma flotante en órdenes de magnitud más rápidos que una CPU convencional. No sólo es más eficiente, los semiconductores han reducido tanto su tamaño y su precio que la computación con GPU es menos costosa. Entre las aplicaciones que se pueden adaptar eficientemente a la GPU, están el álgebra lineal, el procesamiento de imágenes, algoritmos de ordenación y búsqueda, procesamiento de consultas en bases de datos, análisis de finanzas, mecánica de fluidos computacional, predicción meteorológica e inteligencia artificial. Esto es través de

su arquitectura masivamente paralela (Cai et al., 2013, Pallipuram et al., 2012, NVIDIA, 2017).

## Modelos de programación

Las arquitecturas GPGPU de última generación han evolucionado a partir de las GPU, para permitir el procesamiento de tareas computacionales más generales. En la actualidad hay dos principales fabricantes de GPU de propósito general, AMD y NVIDIA®. En noviembre de 2006, NVIDIA® introdujo CUDA®, una plataforma de computación paralela de uso general y un modelo de programación que aprovecha el motor de cálculo paralelo en GPUs NVIDIA para resolver muchos problemas computacionales complejos de una manera más eficiente que en una CPU. Mientras que CUDA se dedica exclusivamente a la computación con GPU de Nvidia, OpenCL™ (Open Computing Language), es mantenido por el grupo Khronos™ y fue hecho para soportar una variedad de arquitecturas como GPUs de diferentes proveedores, DSPs, procesadores Cell y arquitecturas multinúcleo. Las arquitecturas de cálculo OpenCL y CUDA convergen, aunque cada una tiene su propio conjunto de detalles técnicos son conceptualmente idénticas (Pallipuram et al., 2012, Slazynski and Bohte, 2012, NVIDIA, 2016).

Se han estudiado los dos modelos más populares de programación con GPGPU (CUDA y OpenCL) y dos arquitecturas de diferentes proveedores Nvidia y AMD. Los modelos de programación y arquitecturas se compararon para el rendimiento utilizando ANN de reconocimiento de caracteres, que poseen diferentes requisitos de cálculo y comunicación. Se encontró que el modelo de programación CUDA puede extraer más rendimiento de la arquitectura GPU de Nvidia en comparación con OpenCL. La diferencia en el rendimiento se atribuye principalmente a los

diferentes compiladores utilizados por los dos modelos de programación. (Pallipuram et al., 2012, Slazynski and Bohte, 2012).

## Arquitectura CUDA

CUDA es una extensión del lenguaje de programación C y permite a los desarrolladores lanzar y administrar cálculos masivamente paralelos en la GPU. Además, se admiten otros lenguajes, interfaces de programación de aplicaciones o enfoques basados en directivas, como FORTRAN, DirectCompute, OpenACC. Permite incrementos dramáticos en el rendimiento de la computación aprovechando la potencia de la GPU. En CUDA, las tareas se agrupan y transmiten en un conjunto de instrucciones, de la memoria principal del sistema a la memoria de la GPU de tal manera que cada hilo del núcleo funcione con datos diferentes pero ejecute la misma instrucción. En sistemas operativos, un thread o hilo de ejecución es una secuencia de tareas encadenadas muy pequeña que puede ser ejecutada por un sistema operativo. A continuación, ejecutar rutinas del Kernel. Después de eso, los resultados deben ser transferidos de vuelta de la memoria de la GPU a la memoria principal. En algunas arquitecturas emergentes, la GPU y una CPU comparten una memoria (Slazynski and Bohte, 2012, Cai et al., 2013, Thibault and Senocak, 2012, NVIDIA, 2016).

El programa CUDA es un código mixto de GPU y CPU, y aísla el código de la GPU y CPU por el compilador. El código de la GPU (Kernel) se compila dentro de la computación de la GPU en código ensamblador y es ejecutado por el conjunto de multiprocesadores SM de la GPU, cada uno de los cuales consta de varios procesadores escalares (SP). El Kernel se ejecuta en paralelo utilizando un gran número de hilos. Estos hilos se dividen en bloques, cada uno de los cuales se ejecuta en un único multiprocesador SM. Los bloques de hilos suelen tener muchos

más hilos que los procesadores escalares en el multiprocesador, por lo que los bloques de hilos se subdividen en grupos más pequeños ejecutados por los SP. El multiprocesador cambia entre estos grupos, ejecutando los hilos dentro de cada grupo en bloqueo. Todos los hilos ejecutan el mismo programa, pero operan en diferentes partes de los datos (Cai et al., 2013, Fidjeland et al., 2013).

La jerarquía de memoria en CUDA comprende un conjunto de registros (en el chip) y memoria local (residiendo en una DRAM fuera de chip) para cada hilo, una memoria compartida privada para bloques de hilos, una memoria global para todos los hilos iniciados, la arquitectura se muestra en la Figura 10, con una colección de SM de flujo continuo, compuestos por procesadores escalares paralelos SP y una jerarquía de tres niveles de memoria (Pallipuram et al., 2012, Cai et al., 2013).

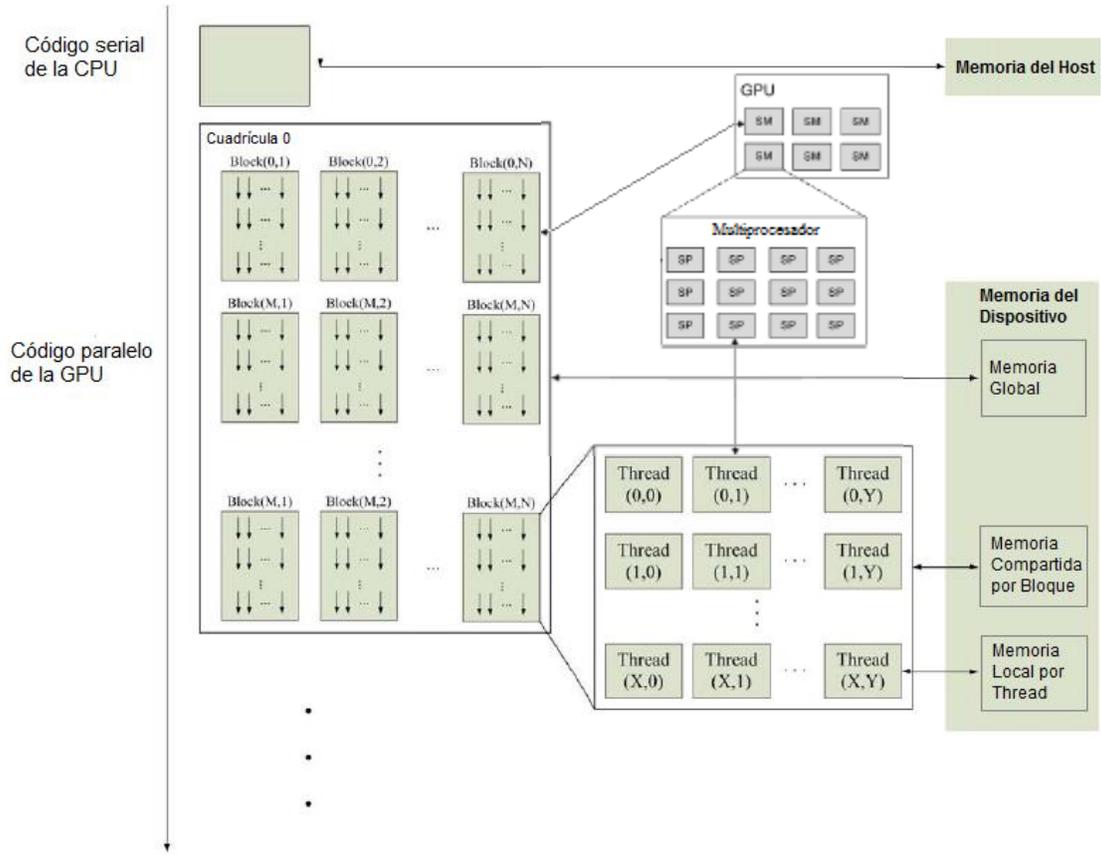


Figura 10. Modelo de programación CUDA GPU (Mei et al., 2014).

Todos los hilos de un bloque en particular tienen acceso a la misma memoria compartida y pueden sincronizarse entre sí, pero sólo pueden ser sincronizados a través del sistema host (CPU). Esta memoria local funciona como un búfer o caché administrado por el programador. Por otro lado, los hilos de bloques diferentes no pueden sincronizarse y pueden intercambiar datos sólo a través de la memoria global de la GPU. Un solo bloque sólo puede contener un número limitado de hilos, dependiendo del modelo del dispositivo. Pero los bloques diferentes se pueden ejecutar en paralelo (Thibault and Senocak, 2012, Fidjeland et al., 2013).

Una ventaja de las GPU modernas es la gran cantidad de ancho de banda de memoria global disponible. Para utilizar este ancho de banda, se basa en patrones de acceso de memoria específicos. Generalmente se debe acceder a la memoria de manera que los grupos de hilos consecutivos en el mismo SM accedan a grupos de palabras consecutivas en la memoria. La unidad más básica de paralelismo es la urdimbre, que es un grupo de 32 hilos. Las instrucciones aritméticas se ejecutan siempre en una urdimbre, y los accesos de memoria se emiten en medio urdimbre (los primeros y últimos 16 hilos de una urdimbre). Los hilos divergen en su ejecución si se toman caminos diferentes en condicionales. Cada urdimbre ejecuta las mismas instrucciones, y en el caso de condicionales cada rama se ejecuta en serie en un subconjunto de los hilos en esa urdimbre, por lo que este último tipo de acceso a la memoria se fusiona (Brette and Goodman, 2012, Fidjeland et al., 2013).

## 2.5 MATLAB®

MATLAB® (MATrix LABoratory; “laboratorio de matrices”) es un paquete de software para computación y visualización numérica de alto rendimiento. La plataforma MATLAB está optimizada para resolver problemas técnicos y de ingeniería. Se utiliza para el aprendizaje de la máquina, manipulación de matrices, implementación de algoritmos, procesamiento de señales e imagen, visión de por computadora, comunicaciones, finanzas computacionales, diseño de control, robótica, y muchos otros tipos de cálculos científicos. MATLAB se encuentra en sistemas de seguridad activa de automóviles, naves espaciales interplanetarias, dispositivos de monitorización de la salud, redes eléctricas inteligentes y redes celulares (MATLAB, 2016b).

Las aplicaciones de MATLAB se desarrollan en un lenguaje de programación propio. Este lenguaje es interpretado, y puede ejecutarse tanto en el entorno interactivo, como a través de un archivo de script (archivos \*.m). Además, MATLAB puede llamar funciones y subrutinas para integrar los algoritmos hechos en MATLAB con aplicaciones externas y otros lenguajes como C, C<sup>++</sup>, Fortran, Java<sup>™</sup> y Microsoft<sup>®</sup> Excel<sup>®</sup> (Hagan et al., 2016, MATLAB, 2016e).

El paquete MATLAB dispone de dos herramientas adicionales que expanden sus prestaciones, a saber, Simulink<sup>®</sup> (plataforma de simulación multidominio) y GUIDE (editor de interfaces de usuario) que es el entorno de desarrollo de GUI (interfaz gráfica de usuario). Además, se pueden ampliar las capacidades de MATLAB con las *cajas de herramientas (toolboxes)* y las de Simulink con los *paquetes de bloques (blocksets)*. Las áreas en las que los *toolboxes* están disponibles incluyen procesamiento de señales, sistemas de control, ANN, lógica difusa, simulación y otras. Una herramienta muy útil es GUI, que nos permiten importar, crear, usar y exportar ANN (Hagan et al., 2016, MATLAB, 2016a).

## Neural Network Toolbox<sup>™</sup>

El *toolbox* de redes neuronales soporta una variedad de arquitecturas de red supervisadas y no supervisadas. Proporciona algoritmos, funciones y aplicaciones para crear, entrenar, visualizar y simular ANN. Puede realizar la clasificación, la regresión, el agrupamiento, la reducción de la dimensionalidad, la predicción de series temporales, el modelado y control de sistemas dinámicos (MATLAB, 2016c).

La caja de herramientas incluye la ANN convolucional y algoritmos de aprendizaje profundo de autocodificador para la clasificación de imágenes y tareas de aprendizaje de características. Para acelerar la formación de grandes conjuntos de datos, puede distribuir cálculos y datos a través de procesadores multinúcleo, GPUs y clústeres de equipos utilizando Parallel Computing Toolbox™ (MATLAB, 2016c).

## Parallel Computing Toolbox™,

El “*Parallel Computing Toolbox*”, permite resolver problemas computacionales y de uso intensivo de datos, utilizando CPUs multinúcleo, GPUs y clústeres de equipos. Las construcciones de alto nivel, los algoritmos numéricos paralelos, los bucles paralelos y los tipos especiales de matrices permiten paralelizar las aplicaciones de MATLAB sin la programación CUDA y los usuarios pueden modificar fácilmente sus códigos. El uso de varios ordenadores puede resolver problemas usando conjuntos de datos demasiado grandes para caber en la RAM de una sola computadora. El único límite para el tamaño del problema es la cantidad de memoria RAM disponible. El uso de MATLAB para el cálculo GPU permite acelerar las aplicaciones con GPUs de forma más fácil que mediante C o Fortran y sin tener que conocer los entresijos de las arquitecturas de las GPUs o las bibliotecas de cálculo GPU de bajo nivel. Esto se logra a través del “*Parallel Computing Toolbox*”, que es compatible con:

- GPUs exclusivamente de NVIDIA CUDA con capacidad de cálculo de la versión 1.3 o posterior, incluidas las GPUs de Tesla.
- Utilización de GPUs directamente desde MATLAB.
  - ◆ Funciones de MATLAB compatibles con GPU como `fft`, `filter` y diversas operaciones de álgebra lineal.
  - ◆ Funciones compatibles con GPU incluidas en los *toolboxes*: Image Processing, Communications System, Neural Network, Phased Array Systems y Signal Processing.

- ◆ Integración del kernel CUDA en las aplicaciones de MATLAB mediante una única línea de código de MATLAB.
- Varias GPUs en estaciones de trabajo y clústers de ordenadores mediante MATLAB workers en “*Parallel Computing Toolbox*” y MATLAB Distributed Computing Server (Hagan et al., 2016, NVIDIA, 2016, MATLAB, 2016e).

Las redes neuronales son algoritmos inherentemente paralelos. El *toolbox*, de computación paralela, cuando se utiliza junto con el *toolbox* de redes neuronales, permite que el entrenamiento y la simulación de las ANNs aprovechen cada modo de paralelismo. Las CPU multinúcleo, GPU y clústers de computadoras con múltiples CPUs y GPUs se pueden combinar para ejecutar los cálculos paralelos utilizando MATLAB Distributed Computing Server™. La forma más sencilla de hacerlo es especificando el entrenamiento y la simulación, utilizando las opciones 'showResources', useParallel' y/o 'useGPU' según sea el caso, es especialmente recomendado para verificar que se está utilizando el hardware esperado. Estas líneas de código utilizan todos los trabajadores (procesadores) disponibles en el grupo paralelo. En algunos casos, podría ser más rápido usar sólo GPUs (MATLAB, 2016c, MATLAB, 2016d).

Ahora se puede entrenar y simular una ANN con datos divididos por muestras entre todos los trabajadores. Sin embargo, es posible realizar esta división manualmente si el tamaño del problema es demasiado grande para el equipo. Se pueden distribuir los datos con diferentes números de muestras por trabajador. Esto se llama equilibrio de carga. Una restricción es que si la red tiene la función de entrenamiento por defecto *trainlm*, verá una advertencia de que los cálculos con GPU no admiten el entrenamiento Jacobiano, sólo el entrenamiento con gradiente. Así que la función de entrenamiento se cambia automáticamente a la función de entrenamiento de gradiente *trainscg*. Para evitar la notificación, puede especificar la función antes del entrenamiento (Hagan et al., 2016).

### III MATERIALES Y MÉTODOS

Este proyecto se realizó en el Laboratorio de Instrumentación Nuclear de la Unidad Académica de Estudios Nucleares (UAEN) de la Universidad Autónoma de Zacatecas. En el Laboratorio de Medidas Neutrónicas (LMN) del Departamento de Ingeniería Nuclear de la Escuela Técnica Superior de Ingenieros Industriales (ETSII) de la Universidad Politécnica de Madrid (UPM) de forma coordinada con el Laboratorio de Patrones Neutrónicos (LPN) perteneciente al Laboratorio de Metrología de Radiaciones Ionizantes (LMRI) del Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (CIEMAT) en Madrid, España.

Para la realización de este proyecto de tesis, el trabajo se dividió en cuatro etapas, la primera etapa se basa en la obtención de datos de espectros de neutrones en la UAEN UAZ, en el LMN UPM y en el LPN CIEMAT. En la segunda etapa se eligieron varias topologías de red correspondientes al diseño de una red neuronal óptima, para posteriormente en la tercera etapa realizar entrenamientos y pruebas con una ANN, implementada en CPU y GPU, a partir de los datos de neutrones obtenidos por la UAEN. La cuarta etapa fue la reconstrucción y obtención de los espectros de neutrones con los datos obtenidos en los laboratorios de la UAEN UAZ, en el LMN UPM y en el LPN CIEMAT, simulados en la ANN entrenada en CPU, y GPU NVIDIA.

#### 3.1 Obtención de las tasas de conteo en la UPM y el CIEMAT

La primera adquisición fue a partir de los datos tratados con el código MNCP-4C por la UAEN, obtenidos originalmente de la base de datos "*Compendium of Neutron Spectra and Detector Responses for Radiation Protection Purposes*" del Organismo Internacional de Energía Atómica (OIEA o IAEA por su siglas en inglés), que incluyen espectros de fuentes isotópicas de referencia, espectros de neutrones de

operación, obtenidos de aceleradores, reactores nucleares y espectros obtenidos de funciones matemáticas. Donde se tomó un conjunto de 211 espectros de neutrones mostrados en unidad de letargia y divididos en 31 grupos de energía dentro del rango de  $1 \times 10^{-9}$  MeV a 630 MeV, al ser tratados en la UAEN, fueron normalizados y divididos en 31 grupos, convertidos a unidades de energía, de los que sólo se eligieron 194 espectros de neutrones y los primeros 27 grupos de energías (Hernandez-Davila et al., 2014, IAEA, 2001, Vega-Carrillo et al., 2007).

La segunda adquisición de datos fue obtenida experimentalmente durante la estancia de investigación en la UPM y el CIEMAT. Para el propósito de determinar el espectro de neutrones, se obtuvieron las tasas de conteo con el detector de centelleo de  ${}^6\text{LiI}(\text{Eu})$ , colocado en el centro de cada una de las esferas moderadoras del BSS de  $0.95 \pm 0.016$  g/cm<sup>2</sup> de densidad, para las fuentes de  ${}^{241}\text{AmBe}$  de 3 Ci y  ${}^{252}\text{Cf}$  de la UPM así como la fuente de  ${}^{252}\text{Cf}$  del CIEMAT. Todas las mediciones se hicieron con el mismo detector y BSS, pertenece al LMN de la UPM.

La sala del LMN de la UPM es una sala tipo bunker con dimensiones de 9 m de ancho  $\times$  16 m de largo  $\times$  8 m de altura y paredes de hormigón de 50 cm de espesor, con una bancada colocada en el centro del piso de la sala, donde se coloca la fuente mediante un sistema de transferencia neumática con sistema de control automatizado y medidas de seguridad adicionales. La fuente de neutrones se coloca a 3 m sobre el suelo, a 4.5 m de las paredes más cercanas, de tal manera que la fuente al momento de hacer las mediciones respectivas quede justo en el centro de la sala y así obtener mejores resultados. En esta sala se realizaron mediciones con dos fuentes (Méndez-Villafañe et al., 2010, Gallego et al., 2004).

La primera fue con la de  ${}^{241}\text{AmBe}$  de 3 Ci, que tenía una tasa de emisión inicial de  $6.64 \times 10^6$  n/s, con fecha de fabricación el 5 de febrero de 1969. Fue colocada a una

distancia de 115 cm entre la fuente y el detector, con un ángulo de 90°. Se hicieron mediciones con las 7 esferas. La segunda fue con la fuente de  $^{252}\text{Cf}$ . Las cuentas de esta fuente fueron tomadas en mayo del 2014 a una distancia de 80 cm entre la fuente y el detector, con un ángulo de 90°. Al finalizar con las fuentes se hicieron las mediciones del fondo con las fuentes de  $^{241}\text{AmBe}$  en la cuba moderadora, en la Figura 11 se muestra la esfera Bonner de 10" sobre la bancada en la sala del Laboratorio de Medidas Neutrónicas de la UPM.

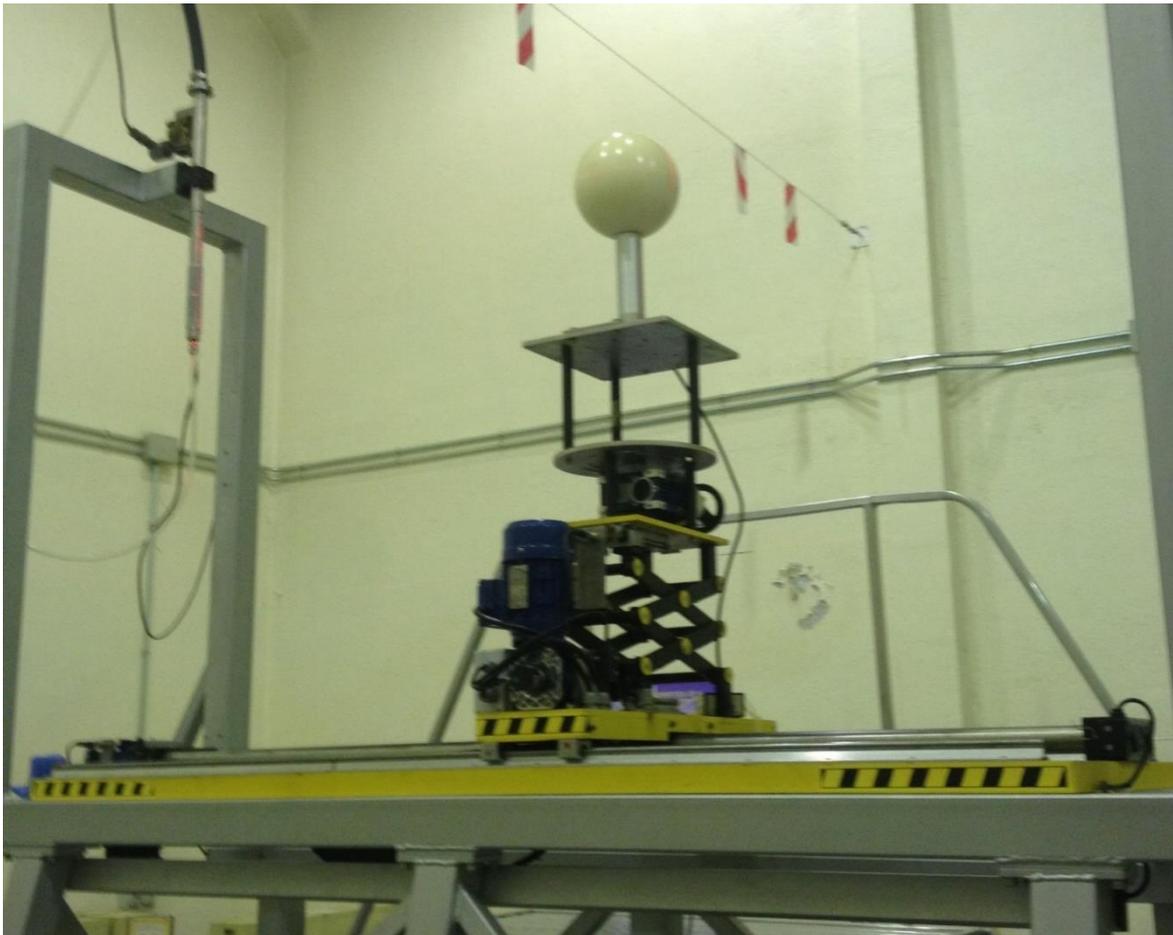


Figura 111. Esfera Bonner de 10" sobre la bancada en LMN de la UPM.

El LPN tiene una sala de (9 m × 7.5 m × 8 m) con paredes de hormigón de 150 cm de grosor. En el suelo, hay una piscina llena de agua de (150 cm × 100 cm × 150

cm) con paredes revestidas de polietileno de alta densidad. En la parte superior, la piscina tiene una puerta de hormigón. La piscina se utiliza para almacenar las fuentes de neutrones. La sala de calibración tiene un banco de acero inoxidable de 250 cm de alto por 375 cm de largo, con una placa de irradiación en la que el dispositivo a calibrar está fijo con respecto al centro de la fuente. Para reducir la dispersión de neutrones en la posición de irradiación, el centro de la fuente se encuentra aproximadamente en el centro de la sala de calibración, estando 4 m por encima del piso. En esta sala se realizaron mediciones con la fuente de  $^{252}\text{Cf}$  (Guzman-Garcia et al., 2015, Mendez-Villafane et al., 2014).

La fuente de  $^{252}\text{Cf}$  tenía una tasa de emisión inicial de  $5.471 \times 10^8$  n/s, con fecha de fabricación el 20 de mayo del 2012. Fue puesta a una distancia de 115 cm entre la fuente y el detector, con un ángulo de  $90^\circ$ . Se hicieron las mediciones con cono de sombra para este sistema, este procedimiento permite aislar la contribución directa de neutrones que inciden sobre la esfera y cuál es la contribución de neutrones retrodispersados. Además se realizaron mediciones sin cono de sombra, estando la fuente de  $^{241}\text{AmBe}$  en la cuba moderadora. Cabe mencionar que el conjunto de conos es diferente para cada fuente. En la Figura 12 se muestra la esfera de 12 pulgadas con su respectivo cono de sombra sobre la bancada en la sala de radiación del Laboratorio de Patrones Neutrónicos del CIEMAT.

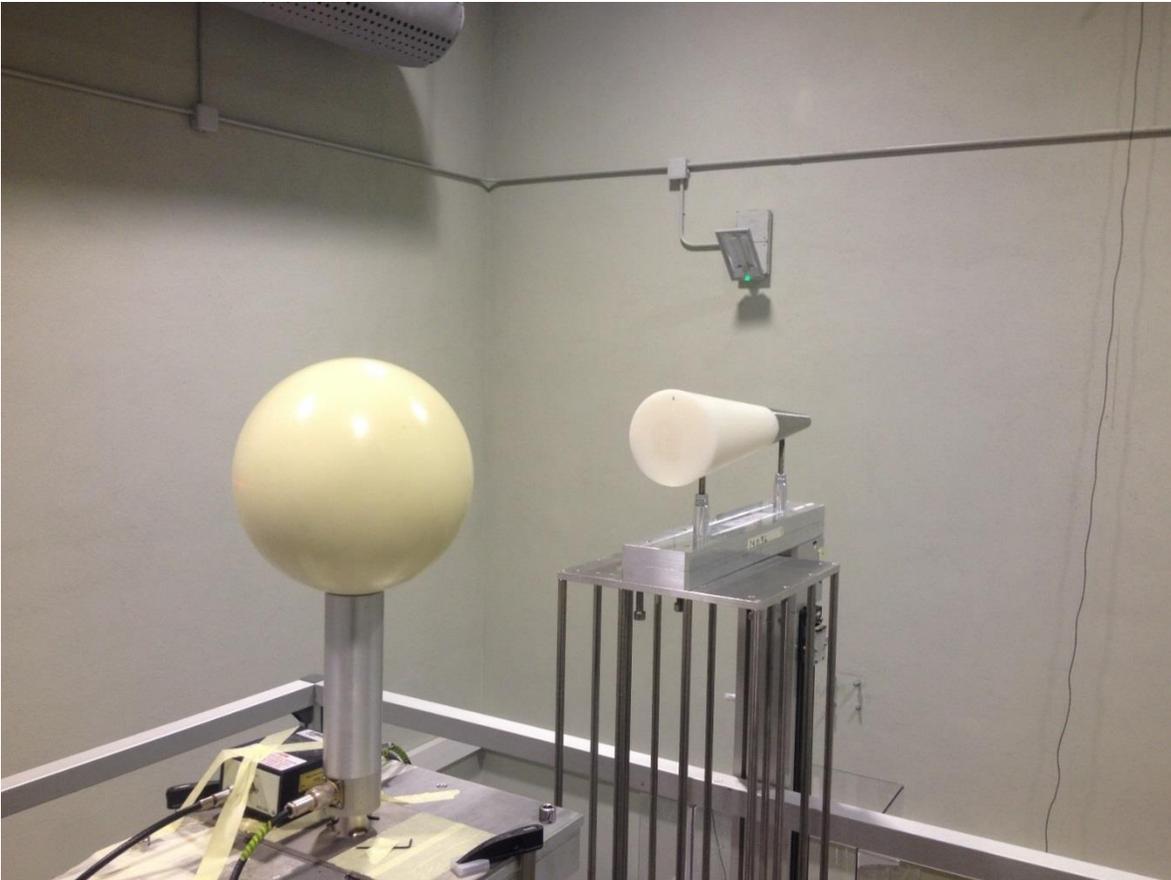


Figura 12. Esfera de 12 pulgadas con cono de sombra, en el LPN del CIEMAT.

Para determinar la radiación debido a los neutrones dispersos se puede seguir la ley del cuadrado inverso que es considerada como fondo. En este caso se calcularon las tasas de conteo experimentalmente para tener un resultado más preciso. Ya que si no se está de acuerdo, se supone que las tasas de conteo se calculan de nuevo (Bramblett and Bonner, 1960).

Cada medición con la fuente de  $^{241}\text{AmBe}$  y  $^{252}\text{Cf}$  da como resultado un espectro de energías de rayos gamma que contienen fotopicos en las energías que corresponden a los primeros canales del software multicanal. Cada pico tiene tres

componentes: primero los rayos gamma procedentes de las interacciones de neutrones con las fuentes isotópicas en los materiales de un blanco; segundo los rayos gamma de fondo; y tercero los neutrones de fondo presentes de forma natural en la sala del laboratorio. Por lo tanto solo se eligió el rango de canales donde aparecen solo los neutrones (Duke et al., 2016).

Las tasas de conteo de las fuentes arrojaron valores mayores de hasta 3 órdenes de magnitud, por lo tanto fue necesario hacer la división de las tasas de conteo de cada fuente entre su respectiva tasa de conteo mayor y así normalizar las entradas para realizar las pruebas necesarias con la ANN. Para los datos de la fuente de  $^{252}\text{Cf}$  obtenidos en la UPM en 2014, las tasas de conteo obtenidas por el BSS fueron menores a la unidad y los valores de salida ya estaban normalizados, por lo tanto no se realizó ningún tratamiento.

### 3.2 Diseño de la red neuronal artificial ANN

La parte experimental de las ANN fue realizada exclusivamente con el código MATLAB<sup>®</sup> 2015a y sus *toolbox* de redes neuronales y computación paralela. El proceso se realizó en dos computadoras de escritorio Hewelett-Packard Company (HP), de dos y cuatro núcleos. La primera con un procesador Pentium(R) Dual-Core CPU E5200 @ 2.50 GHz 2.50 GHz, memoria RAM de 3.0 GB y sistema operativo Windows Vista<sup>™</sup> Home Basic de 32 bits. La segunda con un procesador Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz, memoria RAM de 16.0 GB y sistema operativo Windows 10 pro de 64 bits.

La ANN elegida está conformada por una capa de 7 entradas, dos capas ocultas de 25 neuronas y una capa de salida de 27 neuronas, su arquitectura es 7:25:25:27.

Para la obtención de la ANN elegida se diseñó un algoritmo, el cual hizo la combinación del número de neuronas de cada capa oculta dentro de un rango de 20 a 50 neuronas, así como la combinación de las funciones de transferencia de las capas ocultas usando las funciones tangente, sigmoidea y lineal, después de un gran número de pruebas con diferentes estructuras de red mediante las combinaciones realizadas por el algoritmo diseñado, se eligió la ANN que obtuvo menor error para obtener el espectro de neutrones.

Con base a lo anterior para el diseño de la ANN se fijó la topología *feedforward*, el algoritmo de entrenamiento *trainscg*, dos capas ocultas, el número inicial de neuronas elegido para cada capa oculta fue mayor al número de neuronas de la capa de entrada. Además nunca se cambiaron las entradas con 7 neuronas correspondientes a las 7 tasas de conteo del BSS, sólo se redujeron los niveles de energía a 27 ya que los últimos cuatro niveles fueron cero y la GPU no soportó la matriz. La etapa de entrenamiento puede llevar mucho tiempo con el fin de aproximarse gradualmente a buenos valores de los pesos. El tamaño de los datos de entrenamiento tiene que ser muy grande, por lo tanto, a menudo es casi imposible proporcionar suficientes muestras de entrenamiento.

## Entrenamiento

Para el entrenamiento de la red se tomaron datos de 182 espectros normalizados en unidades de energía de 0 a 1, para este proceso durante la fase se actualizaron los valores de los pesos sinápticos y de los *bias* de tal manera que se obtuvieron valores de salida. El cálculo se realizó al multiplicar los datos de entrada por los pesos sinápticos estos sumados con los valores del *bias* y finalmente multiplicados por la función de transferencia. Se fijó un error mínimo de  $8 \times 10^{-6}$  y un gradiente mínimo de  $1 \times 10^{-8}$  para el rendimiento de la red, además se fijaron  $5 \times 10^5$  iteraciones

con el propósito de llegar al error esperado, y así poder comparar la ANN en las diferentes arquitecturas de hardware. Durante el proceso no se realizaron validaciones, esto mediante la función *dividetrain* que manda todos los datos para el entrenamiento de la ANN, y por lo tanto el entrenamiento de la red concluyó con el criterio de “parado rápido” al alcanzar el menor error fijado.

La función de transferencia utilizada para las capas ocultas de la ANN fue la tangente sigmoidea *tansig* que varía de -1 a 1, y la función de transferencia usada para la salida fue la lineal *pureLin*, para obtener datos concretos. Esta función de transferencia sirve para determinar si la respuesta de un sistema es estable o inestable, o cuan cerca a la estabilidad o inestabilidad está.

## Implementación en MATLAB® con CPU

El diseño, entrenamiento, validación y pruebas de la ANN fueron realizados con el *toolbox* de redes neuronales de en el programa MATLAB® 2015a. De acuerdo con la implementación del algoritmo *trainscg* en una CPU de doble núcleo, la aceleración de un orden de magnitud se puede lograr mediante las tecnologías de aceleración en las operaciones con la matriz de pesos en la plataforma MATLAB. El algoritmo *trainscg* fue considerado una opción favorable para resolver problemas de reconocimiento de patrones, debido a que su requerimiento de memoria es relativamente pequeño y más rápido que los algoritmos estándares de gradiente descendiente. Además, su funcionamiento es considerado para la técnica de “parado rápido” (Hagan et al., 2016, Mei et al., 2014, Møller, 1993).

La misma arquitectura de red se entrenó en la CPU con un solo núcleo, que está fijado por defecto, y en la CPU con los dos núcleos, para poder comparar el

rendimiento de hardware. Las instrucciones para la implementación con CPU y computación distribuida (con varios núcleos o trabajadores) se muestran a continuación. Para abrir un grupo de trabajadores de MATLAB utilizando el perfil de clúster predeterminado, que suelen ser los núcleos de las CPU locales, se utilizó el comando *parpool*, cuando se ejecutó la pantalla mostró dos trabajadores disponibles en la agrupación.

```
pool = parpool
Starting parallel pool (parpool) using the 'local' profile ...
connected to 2 workers.
```

Cabe señalar que la GUI *nntaintool* no aparece cuando se procesa con varios trabajadores de CPU o más de una GPU, por lo tanto, no se puede aplicar el criterio de parada rápido si se desea detener el entrenamiento antes de llegar a los objetivos fijados. Para entrenar la red neuronal con los datos divididos entre los dos trabajadores, se configuró el parámetro de *train* con 'useParallel' 'yes', además para verificar que los cálculos se ejecutaron en varios trabajadores se utilizó el argumento 'showResources', como se muestra a continuación:

```
net1 = train(net,z,t,'useParallel','yes','showResources','yes');
```

## Implementación en MATLAB® con GPU

El estudio actual se centró en el uso de GPU para implementar la ANN *feedforward*. Ya que la CPU todavía no puede cumplir con el requisito de aplicaciones de alta velocidad y en tiempo real, sin embargo trabajando las dos unidades en conjunto, se puede multiplicar el rendimiento computacional en operaciones con matrices. A las GPU que se tuvo acceso fueron dos. La primera fue una tarjeta gráfica NVIDIA®

GeForce® GTX 960 con 1024 núcleos CUDA, velocidad de núcleo de 1127 MHz, memoria de 2048 MB GDDR5 e interfaz de 128 bits, ducto PCI PCI-Express 3.0 x16 y ancho de banda de 112 GB por segundo. Instalada en la CPU de dos núcleos. La segunda fue una tarjeta gráfica NVIDIA® GeForce® GT 720 con 192 núcleos CUDA, memoria de reloj 1800 MHz efectiva, memoria de 2048 MB GDDR5 e interfaz de 64 bits y ancho de banda de 14.4 GB por segundo. Instalada en la CPU de dos núcleos. Por lo tanto, en este trabajo las tecnologías de aceleración de hardware se adoptan para mejorar aún más el rendimiento computacional de las ANN con el algoritmo SCG en GPU, incluso para aplicaciones en tiempo real de alta velocidad. El procesamiento de la canalización a base de GPU se utiliza para reducir el tiempo de procesamiento y el rendimiento de la ANN (Oh and Jung, 2004).

Cabe señalar que el algoritmo de entrenamiento *trainlm* al momento que se implementó la red en la GPU cambia automáticamente a *trainscg* ya que el GPU no soporta el entrenamiento Jacobiano, sólo entrenamiento de gradiente, mientras que el algoritmo *trainbr* al momento que se ejecuta en la GPU no realiza ningún proceso. El código donde se especificaron los objetivos durante el entrenamiento son los mismos que el procesamiento con la CPU. Sin embargo difieren según el procesador que se está utilizando, estas instrucciones se muestran a continuación en los diferentes tipos de computación con los procesadores (Hagan et al., 2016).

Sin embargo la computación distribuida con CPU difiere a la computación distribuida con GPU y a la computación con una sola GPU. La misma arquitectura de red se entrenó en tres diferentes combinaciones arquitectónicas de hardware con GPU, además de las dos anteriores con CPU. El comando que devolvió el número de GPUs en el sistema es `gpuDeviceCount`, y el resultado fue 1, además se pudieron consultar las características de la GPU. Esto incluye su nombre, número de multiprocesadores SM o SIMD de cada multiprocesador y memoria total. Esto mediante el comando `gpuDevice(1)`. Para aprovechar de una forma más sencilla

la GPU se especificó el `train` y el `sim` con el argumento de parámetro `'useGPU'` establecido en `'yes'` (`'no'` es el valor predeterminado).

```
net1 = train(net,z,t,'useGPU','yes');
```

Para la computación distribuida con CPU y la GPU, se especificaron `train` y `sim`, utilizando el grupo paralelo determinado por el perfil de clúster que utiliza. La opción `'showResources'` fue especialmente recomendada en este caso, para verificar que se estaba utilizando el hardware esperado:

```
net1 = train(net,z,t,'useParallel','yes','useGPU','yes',  
'showResources','yes');
```

Estas líneas de código utilizan todos los trabajadores disponibles en el grupo paralelo. Un trabajador para cada GPU única emplea esa GPU, mientras que otros trabajadores operan como CPUs. En algunos casos, podría ser más rápido usar únicamente GPU. En estos casos, se pudo especificar que el `train` y el `sim` utilizaron sólo al usuario con GPU única.

```
net1=train(net,z,t,'useParallel','yes','useGPU','only');
```

Al igual que con la computación distribuida simple (con CPU), la computación GPU distribuida pudo beneficiarse de los valores compuestos manualmente creados. La definición de los valores compuestos por sí mismo, permitió indicar qué trabajadores utilizar y qué trabajadores utilizan las GPU (Hagan et al., 2016).

Para la obtención de resultados con la ANN de la cual se obtuvo un mejor desempeño se realizaron las simulaciones a partir de los datos tratados obtenidos en la UAZ, UPM y el CIEMAT, para así obtener el espectro de neutrones normalizado, para esto se cargó en MATLAB la ANN entrenada, las tasas de conteo normalizadas como entrada a la ANN y para finalizar se realizó la simulación.

### 3.3 Validación y pruebas de la ANN

Para este proceso se tomaron datos de 12 de espectros de neutrones en unidades de energía normalizadas de 0 a 1, se eligieron los más significativos y de diferentes grupos, en el proceso de validación tales datos se compararon con los datos de salida calculados con la ANN de acuerdo a los valores de entrada, tomando como referencia que no hubiera diferencia estadística se realizó la prueba de Chi Cuadrada ( $\chi^2$ ), ya que esta prueba nos permite hacer una decisión de un supuesto o conjetura (hipótesis  $H_0$ ) sobre la población, partiendo de la información muestral de la misma. La validación se realizó por separado del entrenamiento. Lo que se tomó como pruebas fueron las tres fuentes isotópicas de neutrones de la UPM y el CIEMAT, de las que solo se tenía las 7 tasas de conteo y el error, además de que se normalizaron como en los procesos anteriores para obtener la salida deseada. Una vez obtenida la ANN entrenada y validada se realizó la simulación de las tasas de conteo y la salida se comparó con una ANN entrenada con los 194 espectros de neutrones realizando la prueba de Chi Cuadrada ( $\chi^2$ ).

$$\chi^2 = \sum \frac{(E_{ANN} - E_{original})^2}{E_{original}} \dots\dots\dots 5$$

## IV RESULTADOS Y DISCUSIÓN

### 4.1 Tasas de conteo obtenidas en la UPM y el CIEMAT

De las mediciones realizadas con las fuentes de neutrones se realizaron las operaciones para obtener la tasa de conteo en cuentas por segundo y su respectivo error, en la UPM se obtuvieron los datos que se muestran en la Tabla 4, y en la Tabla 5 se muestran los datos obtenidos de la fuente de  $^{252}\text{Cf}$  del CIEMAT, el comportamiento se muestra en la Figura 13.

Tabla 4. Tasa de conteo de las fuentes de neutrones de la UPM.

<b>Diámetro de la esfera</b> [pulgadas]	<b>Fuente de <math>^{252}\text{Cf}</math> de la UPM, canales del 712 al 1381</b>	<b>Fuente de <math>^{241}\text{AmBe}</math> de la UPM, canales del 200 al 4094</b>
	Tasa de conteo [cuentas/seg]	Tasa de conteo [cuentas/seg]
<b>0</b>	$1.84 \times 10^{-3} \pm 2.44 \times 10^{-4}$	$6.06 \times 10^{-1} \pm 5.48 \times 10^{-3}$
<b>2</b>	$1.68 \times 10^{-2} \pm 4.07 \times 10^{-4}$	$4.78 \times 10^{-1} \pm 1.86 \times 10^{-2}$
<b>3</b>	$5.33 \times 10^{-2} \pm 1.04 \times 10^{-3}$	$1.49 \pm 3.01 \times 10^{-2}$
<b>5</b>	$1.63 \times 10^{-1} \pm 1.94 \times 10^{-3}$	$5.49 \pm 7.41 \times 10^{-2}$
<b>8</b>	$1.74 \times 10^{-1} \pm 2.31 \times 10^{-3}$	$8.16 \pm 8.81 \times 10^{-2}$
<b>10</b>	$1.28 \times 10^{-1} \pm 1.26 \times 10^{-3}$	$7.33 \pm 7.91 \times 10^{-2}$
<b>12</b>	$8.45 \times 10^{-2} \pm 1.01 \times 10^{-3}$	$6.24 \pm 6.19 \times 10^{-2}$

Tabla 5. Tasa de conteo de la fuente de neutrones  $^{252}\text{Cf}$  del CIEMAT.

<b>Diámetro de la esfera</b>	<b>Fuente de <math>^{252}\text{Cf}</math> del CIEMAT</b>	<b>Fuente de <math>^{252}\text{Cf}</math> del CIEMAT, menos fuente con cono Canales del 6001 al 1600</b>
[pulgadas]	Tasa de conteo [cuentas/seg]	Tasa de conteo [cuentas/seg]
<b>0</b>	24.71±0.21	24.71±0.21
<b>2</b>	49.67±0.44	13.12±0.56
<b>3</b>	129.85±1.09	70.27±1.21
<b>5</b>	312.45±2.48	231.01±2.57
<b>8</b>	331.86±1.76	278.17±1.82
<b>10</b>	242.72±1.89	208.97±1.92
<b>12</b>	165.80±1.36	146.07±1.37

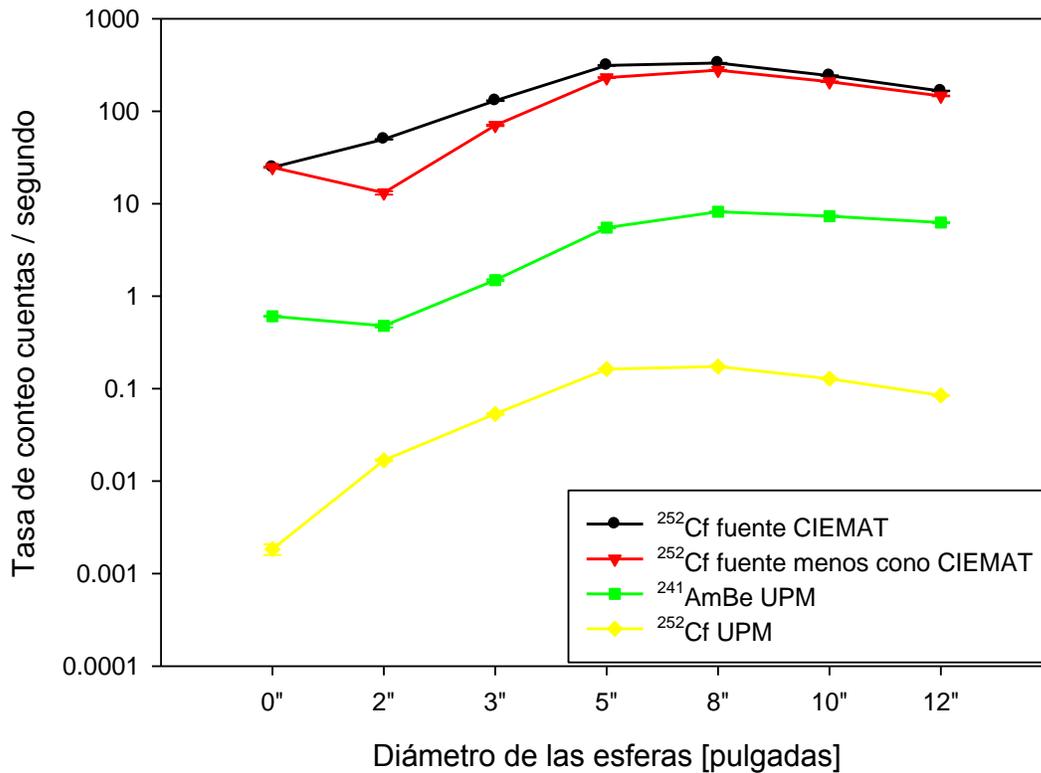


Figura 13. Comportamiento tasas de conteo de diferentes fuentes de neutrones.

Se puede ver en la figura anterior, que la forma típica de las respuestas esfera Bonner es decir, el máximo en la respuesta se desplaza hacia las esferas de mayor tamaño, como se menciona en el capítulo dos, esto se debe principalmente al creciente número de colisiones de neutrones con el moderador, a medida que aumenta el espesor de moderador, en otra parte, en el proceso de absorción que se produce por el neutrón en el moderador antes de que tenga la oportunidad de alcanzar el detector, además de esto el comportamiento también es así, porque en la mayoría de los espectros de neutrones tienen mayor flujo de neutrones en las esferas de mayor tamaño para este BSS.

## 4.2 Diseño y entrenamiento de la ANN en CPU y GPU

Después de realizar varios entrenamientos con diferentes topologías de red, la arquitectura que proporcionó los mejores resultados fue 25:25:27. Ya que al introducir tres capas ocultas, a la salida de la ANN se obtenía mucho ruido y el proceso era más lento, además lo mismo pasaba cuando se tenían capas de diferente número de neuronas. En la Tabla 6 y Tabla 7 se muestra el rendimiento durante el entrenamiento que se realizó en CPU con un sólo núcleo, en CPU con todos los núcleos en cada ordenador, en CPU con todos los núcleos y GPU, en CPU con un sólo núcleo y GPU, además del entrenamiento sólo en GPU. Cabe señalar que el entrenamiento de la ANN se generalizó para todos los grupos de espectros de neutrones.

Tabla 6. Rendimiento de las arquitecturas Pentium y GTX 960.

	<b>CPU Pentium con GPU GTX 960</b>			
<b>Error fijado de <math>8 \times 10^{-6}</math></b>	Tiempo minutos	Iteraciones	Memoria CPU MB	CPU %
<b>CPU 1 núcleo</b>	0:28:30	121205	606	98
<b>CPU 2 núcleos</b>	0:24:38	120193	389	47
			429	50
<b>CPU 2 núcleos y GPU</b>	3:18:05	317540	388	48
			427	46
<b>CPU 1 núcleo y GPU</b>	1:28:33	124159	386	48
			426	45
<b>GPU</b>	1:33:17	123473	238	59

Como se puede apreciar en la tabla anterior, la CPU con dos núcleos funcionando es la que obtiene mejor rendimiento en cuanto al tiempo de operación que es lo que principalmente se buscó al comparar las arquitecturas de hardware, en apariencia es más rápido, pero por el contrario se necesita mayor capacidad de memoria. Además de que para esta CPU al utilizar computación paralela, la GUI *nntaintool* no se muestra, ya que la CPU multinúcleo está utilizando todos los recursos y deja de procesar los gráficos.

Tabla 7. Rendimiento de las arquitecturas Intel y GT 720.

	<b>CPU Intel con GPU GT 720</b>			
<b>Error fijado de <math>8 \times 10^{-6}</math></b>	Tiempo	Iteraciones	Uso de memoria CPU MB	Rendimiento en CPU %
<b>CPU 1 núcleo</b>	0:34:26	247001	1082	21
<b>CPU 4 núcleos</b>	0:09:44	94559	284	12.4
			218	12.4
			228	12.4
			284	12.4
<b>CPU 4 núcleos y GPU</b>	0:52:43	149414	284	56
			218	
			228	37
			284	
<b>CPU 1 núcleo y GPU</b>	1:13:24	152409		53
				36
<b>GPU</b>	0:55:06	108463	1082	18

La arquitectura de hardware que presento un mejor rendimiento computacional durante el entrenamiento de la ANN fue la CPU Pentium con los cuatro núcleos, considerando el tiempo de entrenamiento. Además de que realizo el entrenamiento

con menos iteraciones. Como se puede observar el rendimiento se dividió en los cuatro núcleos haciendo más rápido el procesamiento. Esto es porque son núcleos muy grandes y están inmersos dentro de la misma CPU. La GPU ayudó a la CPU al procesamiento de los datos y está muy cerca de obtener el mismo rendimiento de la CPU con un solo núcleo. Cabe señalar que la GPU es una tarjeta para gráficos y ahora se utilizó para propósito general GPGPU.

La simulación de los datos de validación se realizó en la ANN que se entrenó con la CPU Pentium de 4 núcleos, ya que fue la que realizó el cálculo de los pesos sinápticos en un menor tiempo e iteraciones, las ANNs entrenadas en las otras arquitecturas de hardware no se utilizaron porque para todas se fijaron los mismos parámetros, principalmente porque para todas las ANNs se fijó el error mínimo de  $8 \times 10^{-6}$  y todos los entrenamientos con diferente combinación de arquitectura hardware cumplieron con el objetivo. En GUI *nntraintool* que aparece por “default” durante el entrenamiento en la Figura 14, se observa finalizado el proceso con computación paralela.

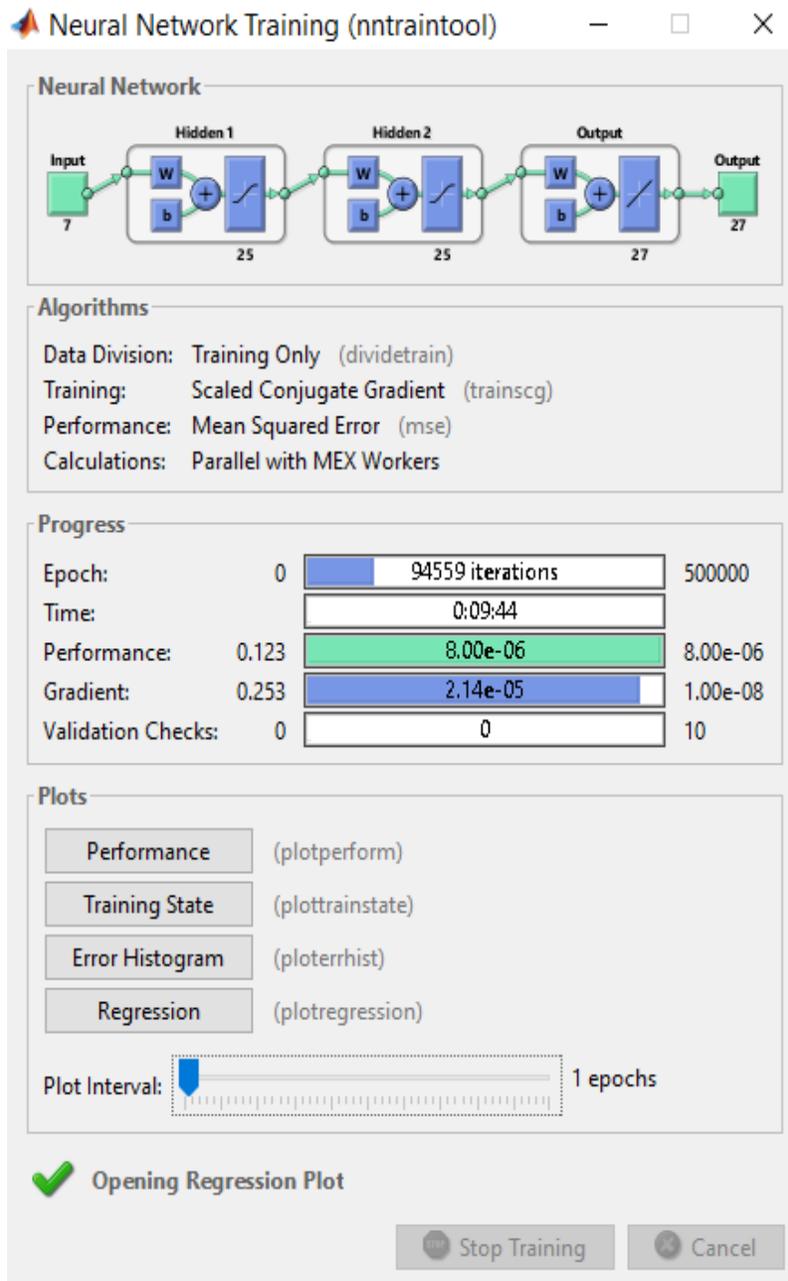


Figura 14. GUI ntraintool en el entrenamiento de computación paralela.

Se establecieron 3 criterios de parado del entrenamiento: cuando llegara a  $5 \times 10^5$  iteraciones, al alcanzar un error cuadrático medio de  $8 \times 10^{-6}$ , cuando el gradiente alcanzara un valor de  $1 \times 10^{-8}$ . De esta manera, el entrenamiento se detuvo a las

94559 iteraciones usando todos los trabajadores de CPU, en un tiempo de 0:09:44, y alcanzó un error mínimo de  $8 \times 10^{-6}$ . En la Figura 15, se muestra cómo disminuyó el gradiente conforme se realizaban las iteraciones, sin embargo el gradiente es muy variante y algunas veces tiende a aumentar cuando encuentra mínimos locales.

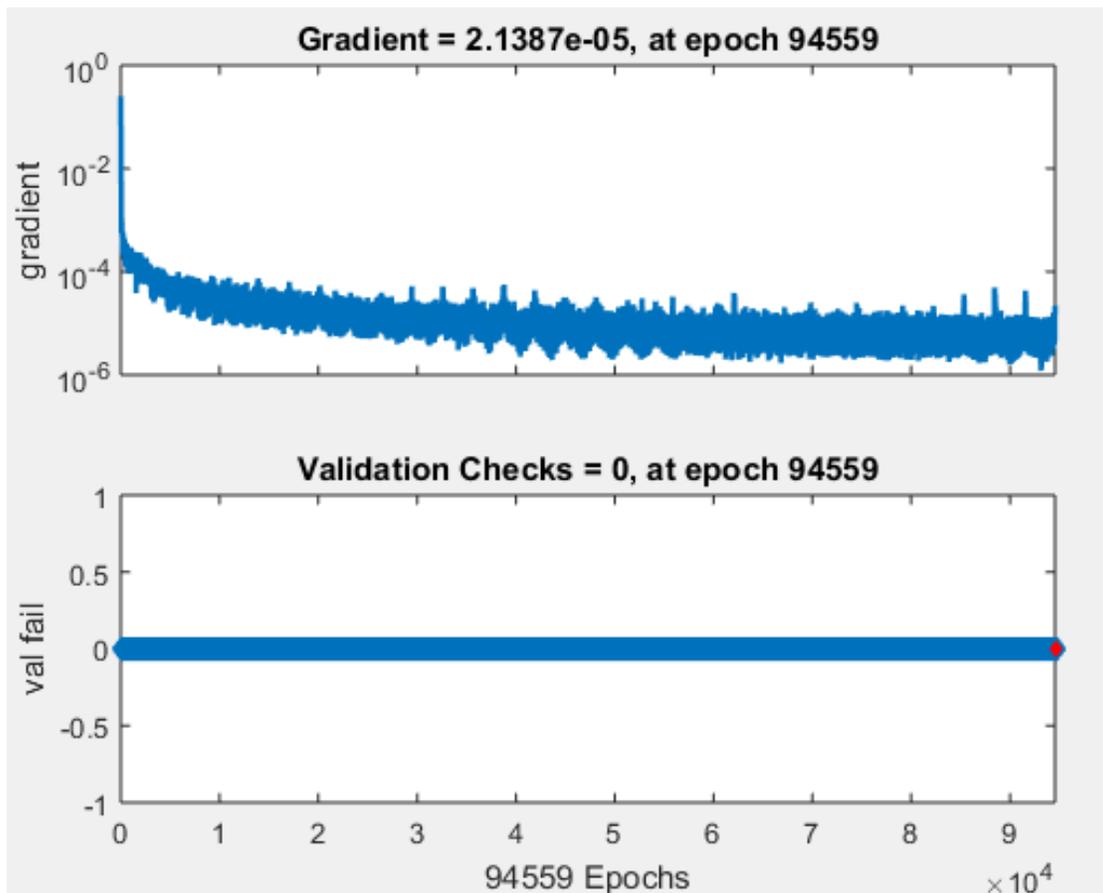


Figura 15. Chequeo de validaciones.

Por lo anterior no se realizaron validaciones dentro del entrenamiento, porque normalmente para este problema, al momento de entrenar la red la validación cumple con el criterio de 6 iteraciones consecutivas sin disminuir el error para detener el entrenamiento y por lo tanto no se llega a un valor óptimo. Así mismo, en la Figura 16 se observa durante el entrenamiento como disminuía el error conforme

se realizaban las iteraciones y transcurría el tiempo, se puede observar que, en la iteración 3000 aproximadamente, el error ya no disminuyó significativamente. Por lo tanto el usuario puede decidir detener el entrenamiento en cualquier momento, mediante la GUI *nntaintool*.

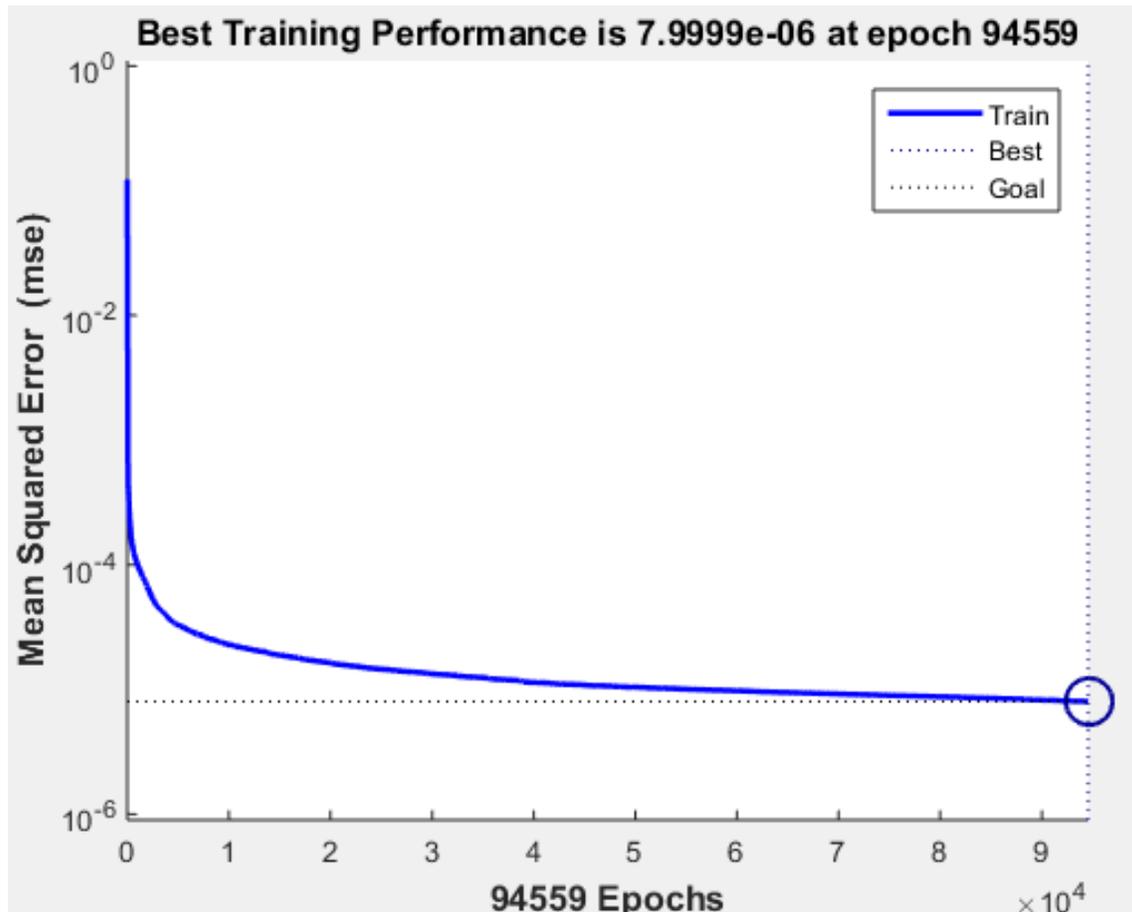


Figura 16. Rendimiento de la red en GPU.

Finalmente, en la Figura 17 se puede apreciar la regresión de los datos durante el entrenamiento. El entrenamiento de la ANN alcanzó una eficiencia del 99.919%. Al no usar datos para validación y pruebas durante el entrenamiento la ANN se obtuvo un error menor y por lo tanto más datos para el entrenamiento. Lo anterior se eligió porque durante el proceso de elección de red se realizaron pruebas y se determinó

que al agregar datos para validaciones, la ANN concluía el entrenamiento en menos de 100 iteraciones y no se obtuvo el error mínimo deseado por el usuario.

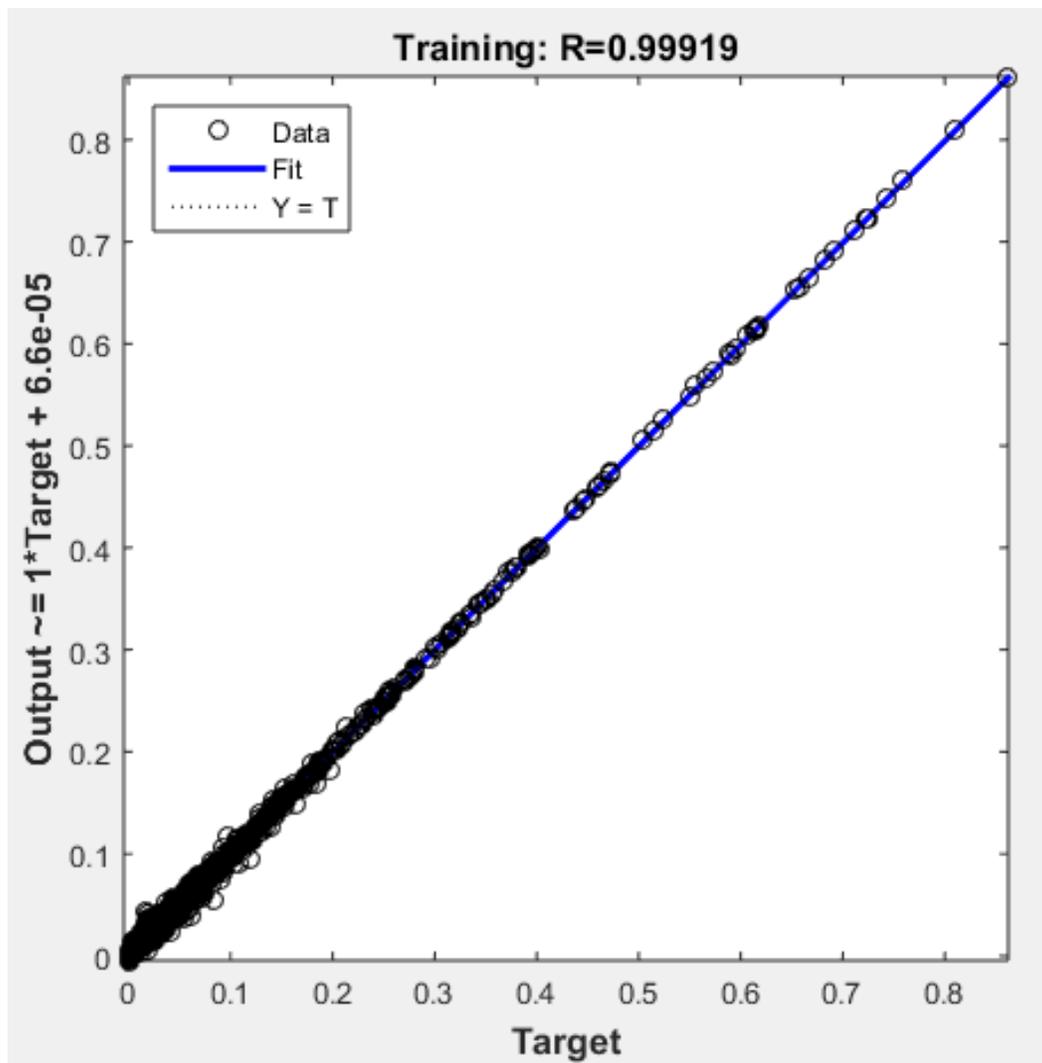


Figura 17. Regresión de datos durante el entrenamiento en GPU.

Para todas las implementaciones de ANN en arquitecturas hardware la regresión fue la misma es por eso que no hay diferencias entre ANN entrenadas.

### 4.3 Validación y pruebas de la ANN

Los resultados de la validación en la ANN entrenada usando los cuatro núcleos de la CPU Pentium, son datos de 12 espectros diferentes entre grupos, en la Tabla 8 se indica el espectro con su respectivo valor de  $\chi^2$ , entre los espectros simulados en la ANN entrenada y los de referencia.

Tabla 8. Resultados de  $\chi^2$  para los espectros de validación.

<b>Espectro</b>	<b>Valor de <math>\chi^2</math></b>	<b>Espectro</b>	<b>Valor de <math>\chi^2</math></b>
ISO D <sub>2</sub> O/ <sup>252</sup> Cf	$2.41 \times 10^{-2}$	CZECH por encima del equipo	$3.58 \times 10^{-3}$
<sup>241</sup> AmBe PTB Cono	$2.81 \times 10^{-3}$	NTL111 a 115cm	$1.72 \times 10^{-2}$
<sup>238</sup> PuBe CERN	$6.39 \times 10^{-2}$	Pollux Arriba	$5.31 \times 10^{-2}$
<sup>238</sup> PuF PNL a 50cm	$2.89 \times 10^{-2}$	PuPlant Loc5	$2.27 \times 10^{-2}$
Trawsfynydd Filtro	$1.39 \times 10^{-2}$	<sup>244</sup> Cm Glovebox	$2.12 \times 10^{-2}$
HinkleyPoint	$4.42 \times 10^{-1}$	Acelerador de Boro Epitermal	$1.25 \times 10^{-1}$

La prueba  $\chi^2$  se considera una prueba no paramétrica que mide la discrepancia entre una distribución observada y otra teórica o esperada; a esta prueba también se le conoce como la prueba de bondad de ajuste. La prueba permite determinar si las diferencias entre los valores de la distribución observada o experimental y la distribución teórica o esperada se deben al azar o las diferencias son significativas. En las siguientes figuras se observa visualmente el ajuste entre los espectros esperados contra los calculados con la ANN mostrados en la tabla anterior.

Se sabe que el  $^{252}\text{Cf}$  es un isótopo artificial que genera neutrones a través de la fisión espontánea en el 3.1 % de los casos y decae emitiendo partículas alfa en un 96.9 %. En este caso se encuentra en una mezcla con agua pesada y emite un espectro diferente al  $^{252}\text{Cf}$  normal, este espectro se muestra en la Figura 18. El agua pesada en conjunto con el  $^{252}\text{Cf}$  y tiene una contribución importante de neutrones de energía intermedia, generados durante la moderación. Sin embargo se sigue observando el máximo, probablemente del californio del  $^{252}\text{Cf}$ .

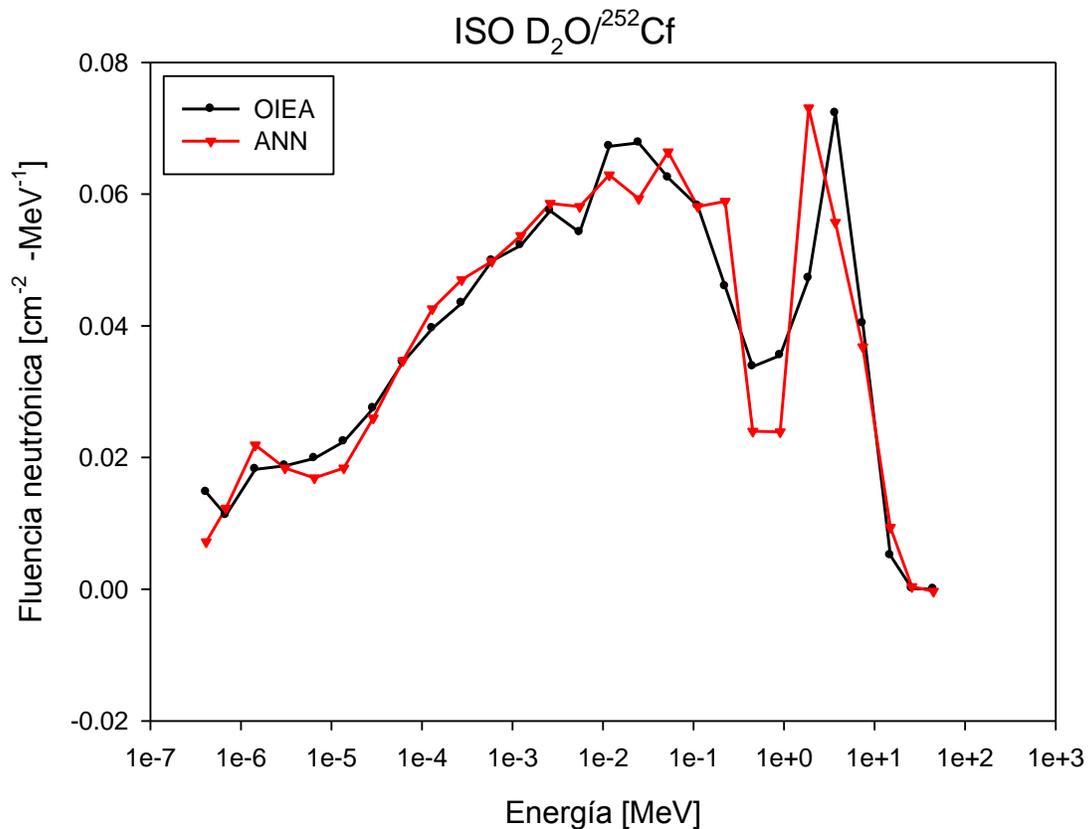


Figura 18. Espectros de neutrones de referencia y por ANN de ISO D<sub>2</sub>O/<sup>252</sup>Cf.

La fuente de  $^{241}\text{AmBe}$  emite neutrones cuya energía varía de 0.1 a 10 MeV y tiene un máximo 4 MeV. Como todos los isótopos del americio son radioactivos y el  $^{241}\text{Am}$  es el isótopo más común del americio, también es el isótopo más prevalente de americio en los residuos nucleares. En la Figura 19, se muestra el  $^{241}\text{AmBe}$  calibrado en PTB con Cono, en el que se puede observar un mejor ajuste entre la salida de la ANN y los el espectro esperado con una  $\chi^2$  de  $2.81 \times 10^{-3}$ , que es el espectro que más se ajusta de los doce elegidos para la validación.

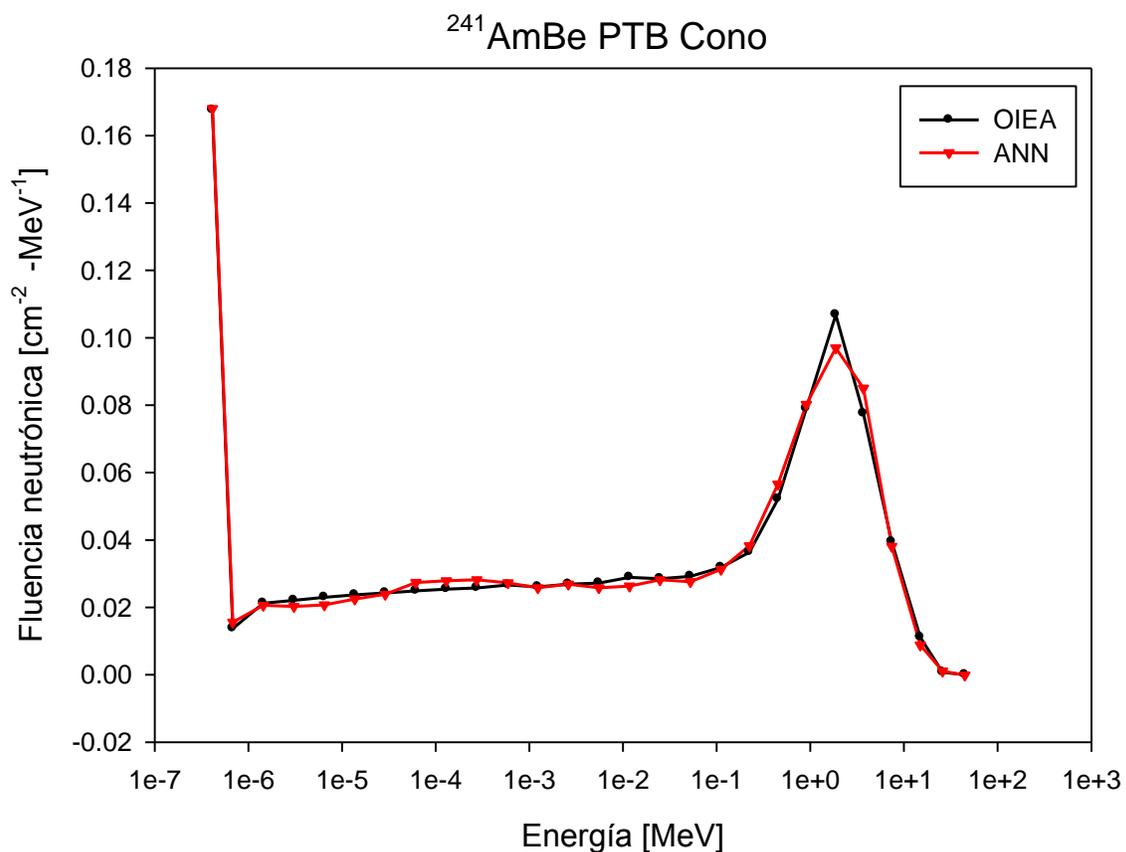


Figura 19. Espectros de neutrones de referencia y por ANN de  $^{241}\text{AmBe}$  PTB Cono.

La fuente de  $^{238}\text{Pu}$  emite partículas alfa que reaccionan con el Be produciendo neutrones, esta situación ocurre en instalaciones donde se procesan materiales nucleares. En la Figura 20 se muestra el espectro de  $^{238}\text{PuBe}$  dividido principalmente en 2 grupos de energía o con mayor fluencia neutrónica, uno con una fluencia baja en la zona de neutrones epitérmicos y una significativa fluencia en el grupo de neutrones rápidos. Y en la Figura 21 también se añadió otro espectro de  $^{238}\text{Pu}$  pero mezclado con F, donde se observa un cambio significativo contra el de la Figura 20.

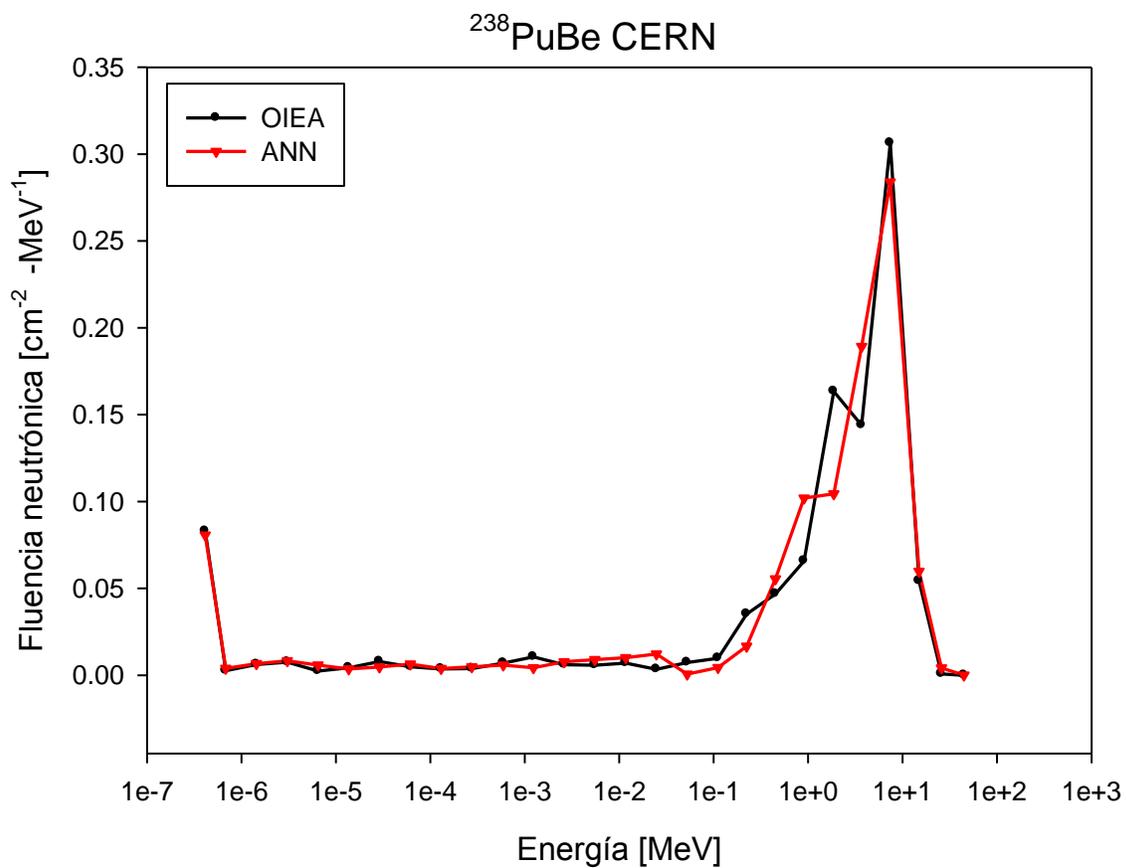


Figura 20. Espectros de neutrones de referencia y por ANN de  $^{238}\text{PuBe}$  CERN.

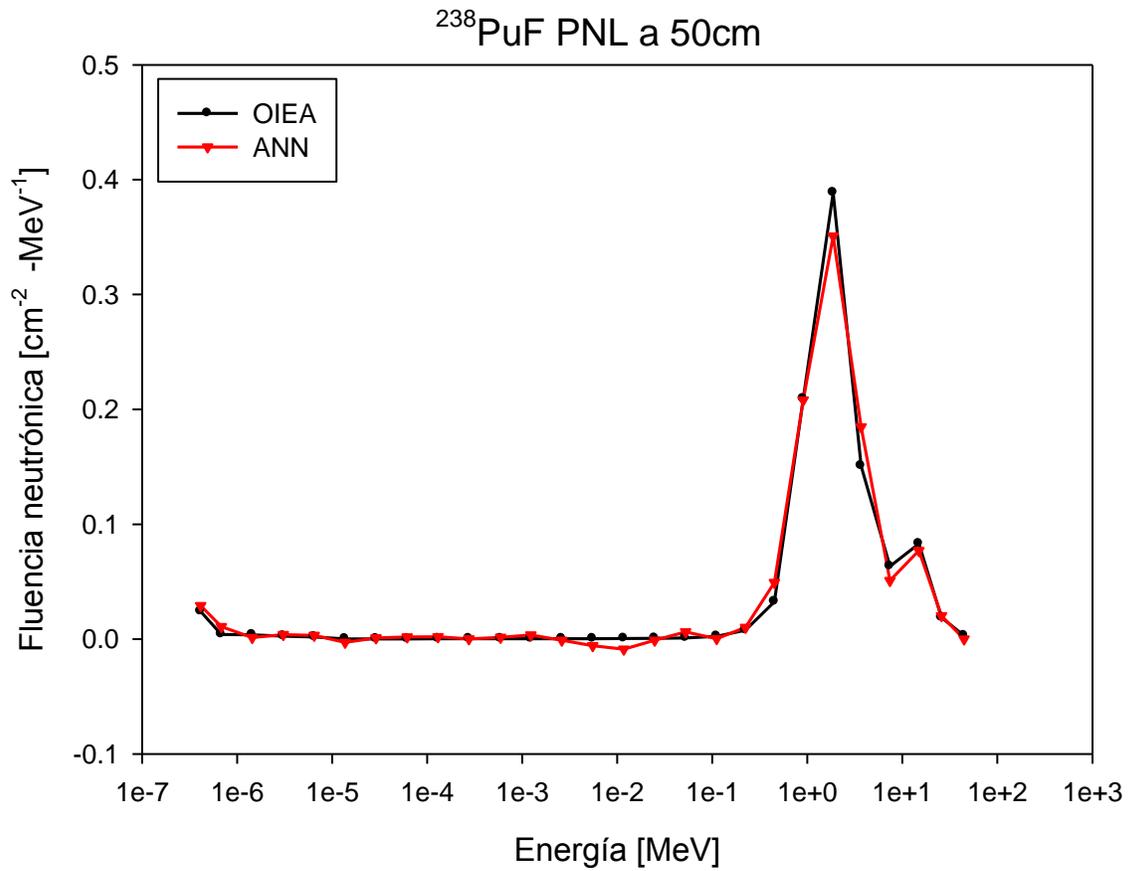


Figura 21. Espectros de neutrones de referencia y por ANN de <sup>238</sup>PuF PNL a 50cm.

En la Figura 22 se muestra un espectro de la central nuclear de Trawsfynydd y el obtenido de la ANN y en la Figura 23 se muestra el espectro obtenido de la central nuclear de Hinkley Point y la ANN, que según la prueba de  $\chi^2$  es el que tiene menor ajuste.

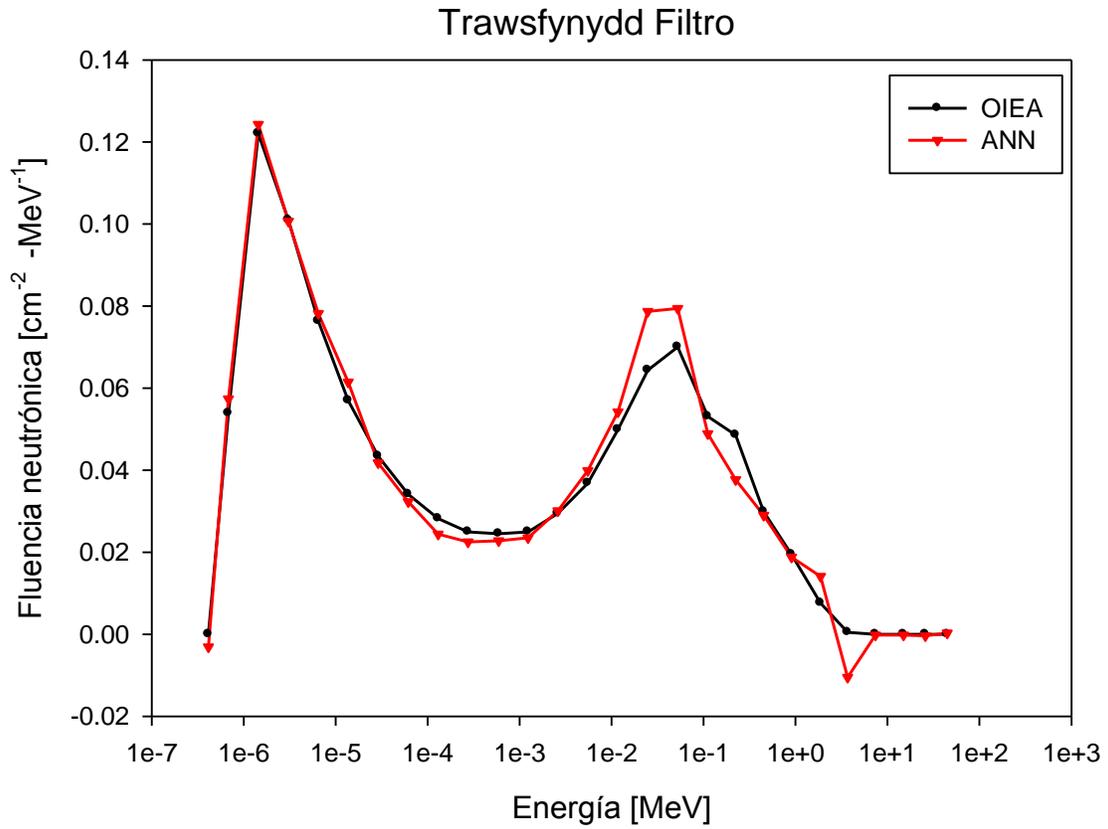


Figura 22. Espectros de neutrones de referencia y por ANN de Trawsfynydd Filtro.

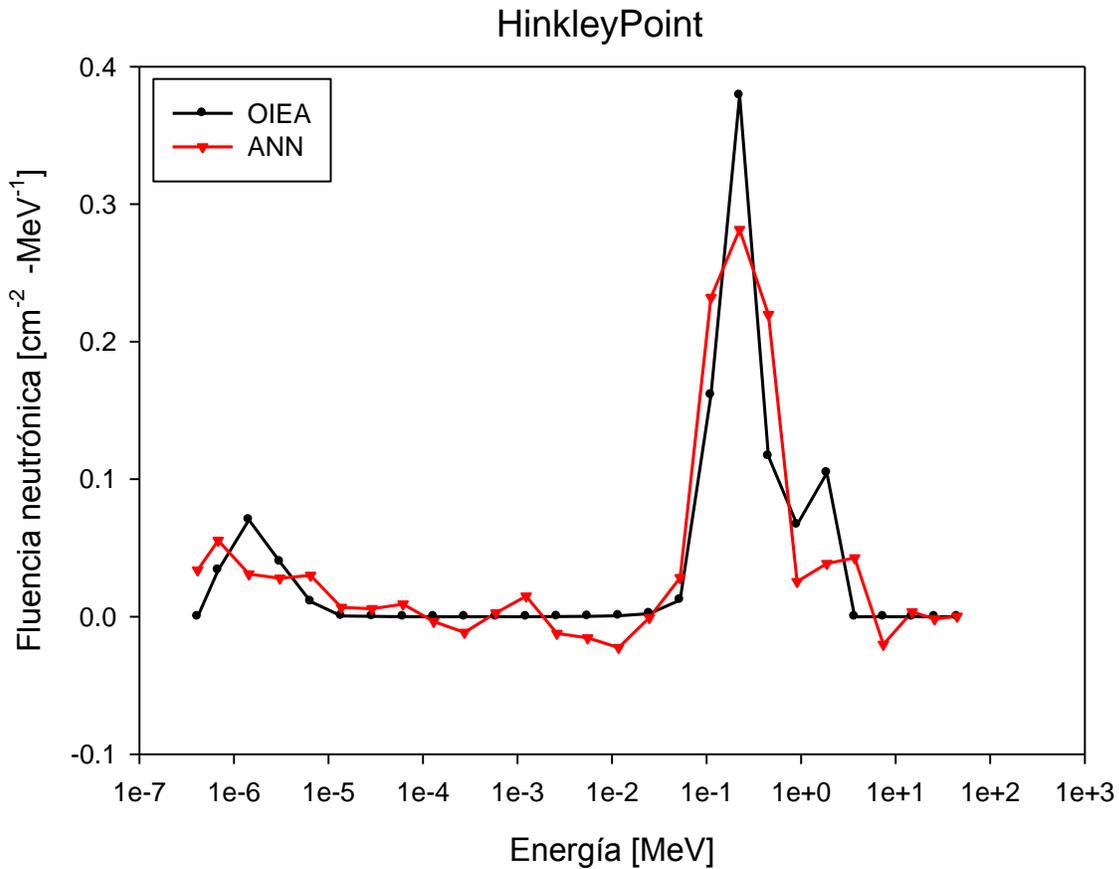


Figura 23. Espectros de neutrones de referencia y por ANN de Hinkley Punto.

En los espectros de la Figura 24 hay una distribución uniforme de energía en los primeros grupos, desde los térmicos hasta los de energía intermedia y una fluencia muy pequeña en los neutrones rápidos y en la Figura 25 se presenta un buen ajuste aunque sea un espectro con mucha variación, aquí se comprueba que la ANN entrenada presenta el error de  $8 \times 10^{-6}$ .

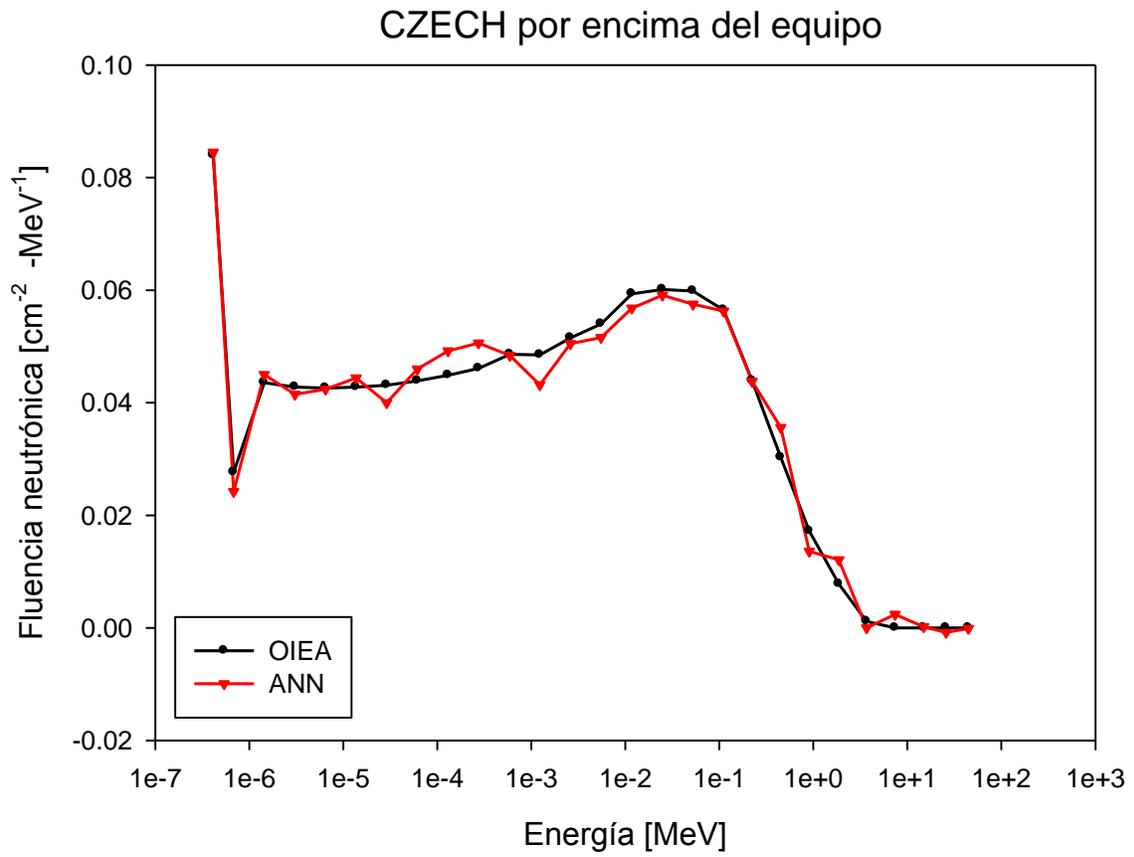


Figura 24. Espectros de neutrones de referencia y por ANN de CZECH por encima del equipo.

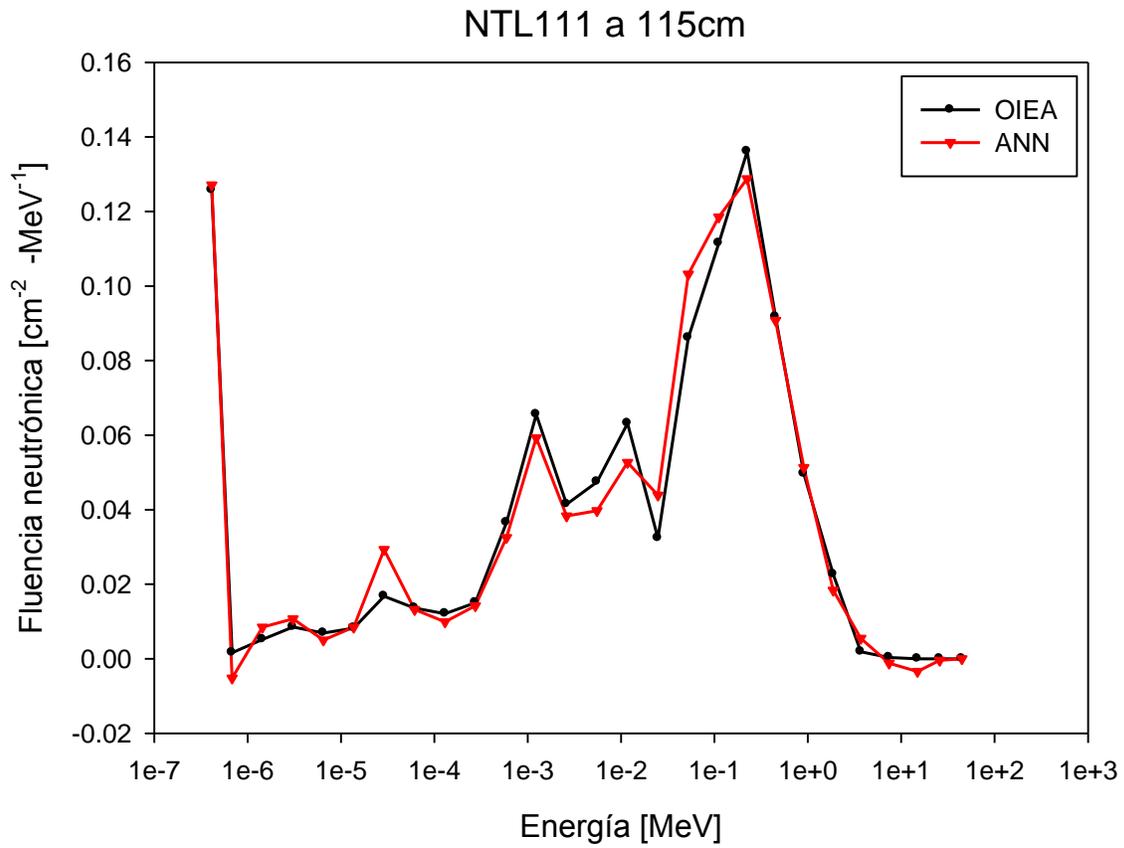


Figura 25. Espectros de neutrones de referencia y por ANN de NTL111 a 115 cm.

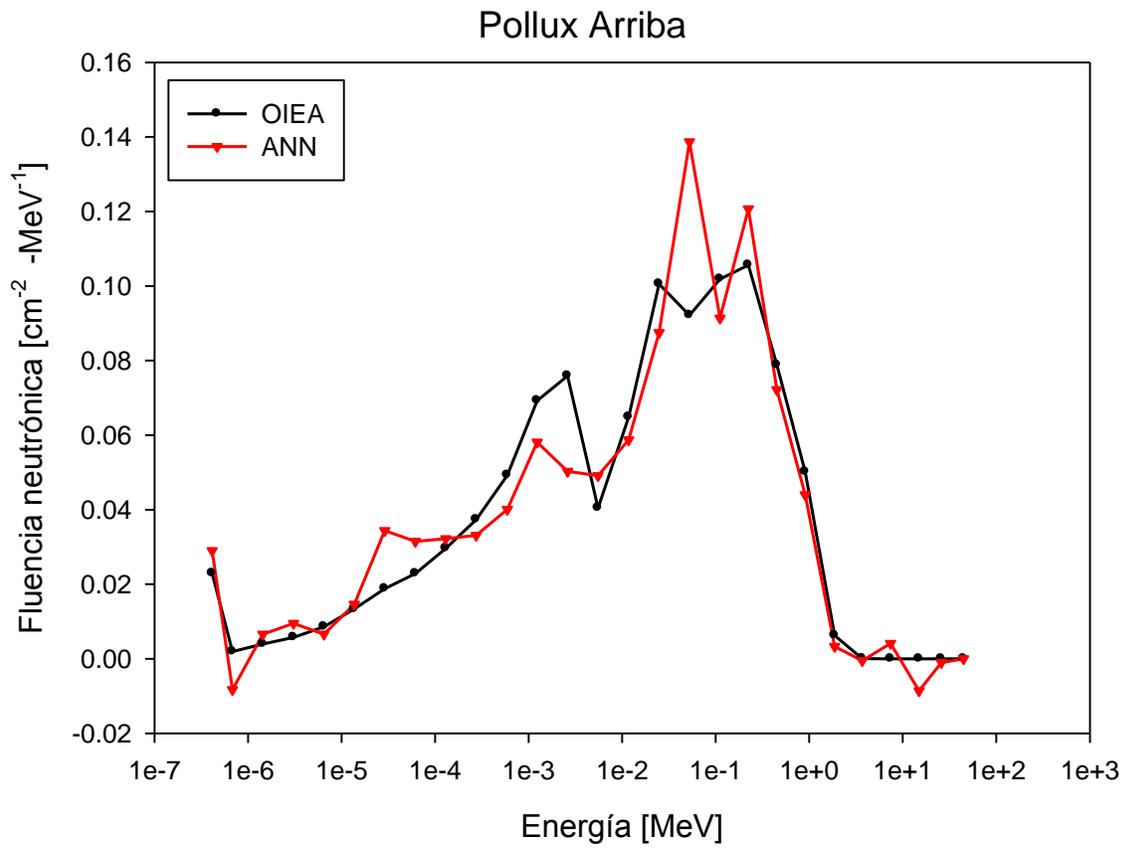


Figura 26. Espectros de neutrones de referencia y por ANN de Pollux Arriba.

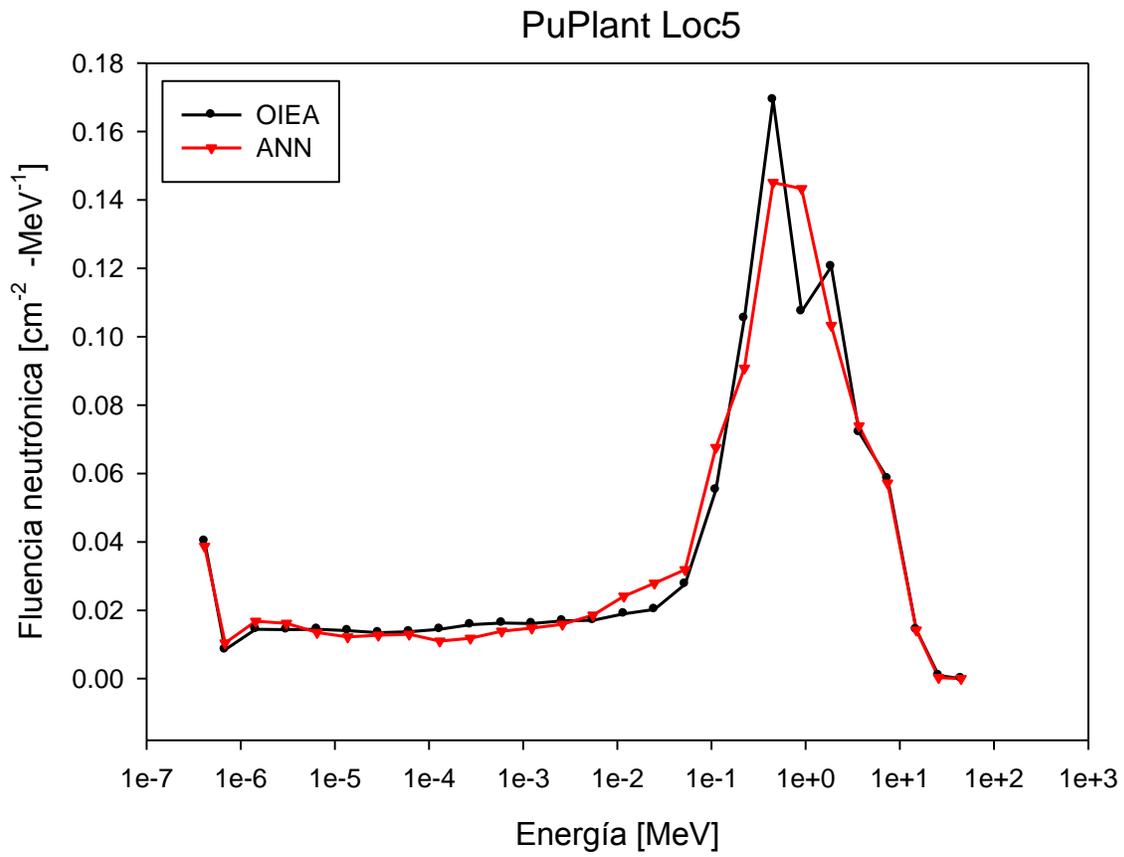


Figura 27. Espectros de neutrones de referencia y por ANN de PuPlant Loc5.

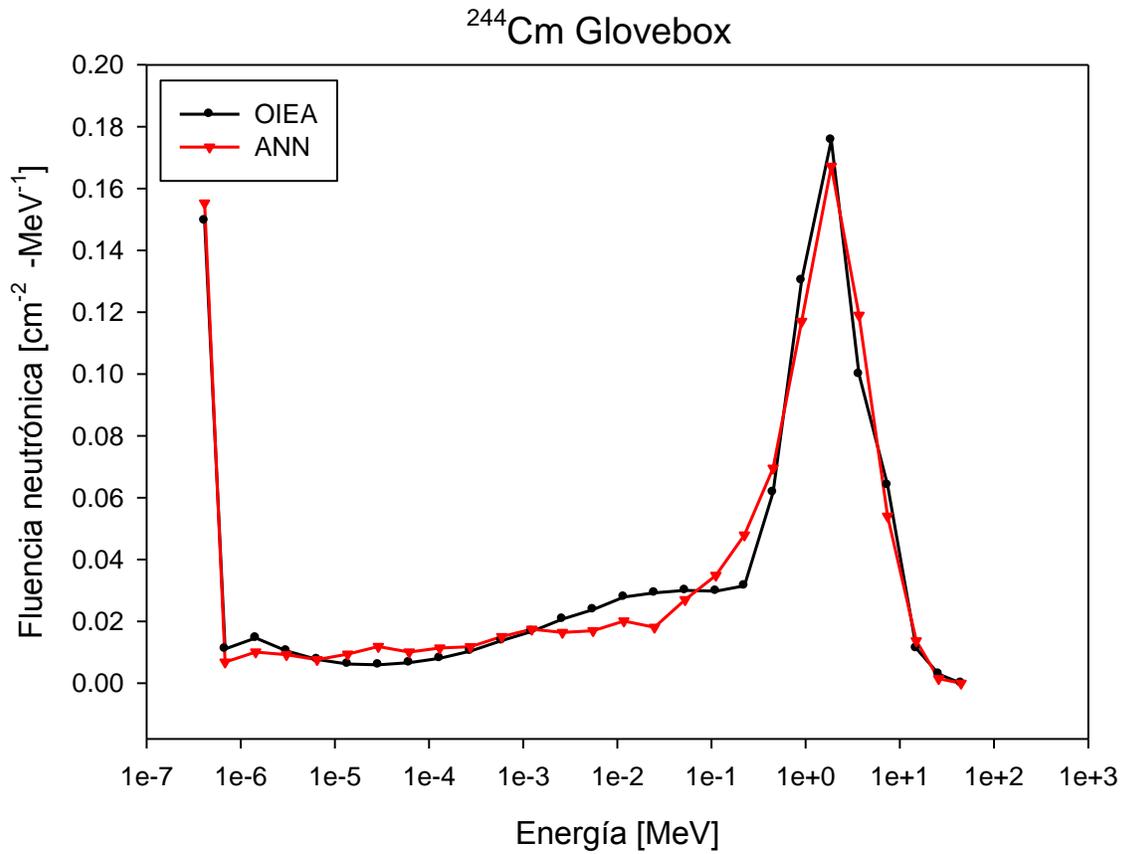


Figura 28. Espectros de neutrones de referencia y por ANN de <sup>244</sup>Cm Glovebox.

En la Figura 29 se muestra el último espectro de los elegidos para validación y con una prueba estadística  $\chi^2$  de  $1.25 \times 10^{-1}$ , sin embargo visualmente tiene mucho error.

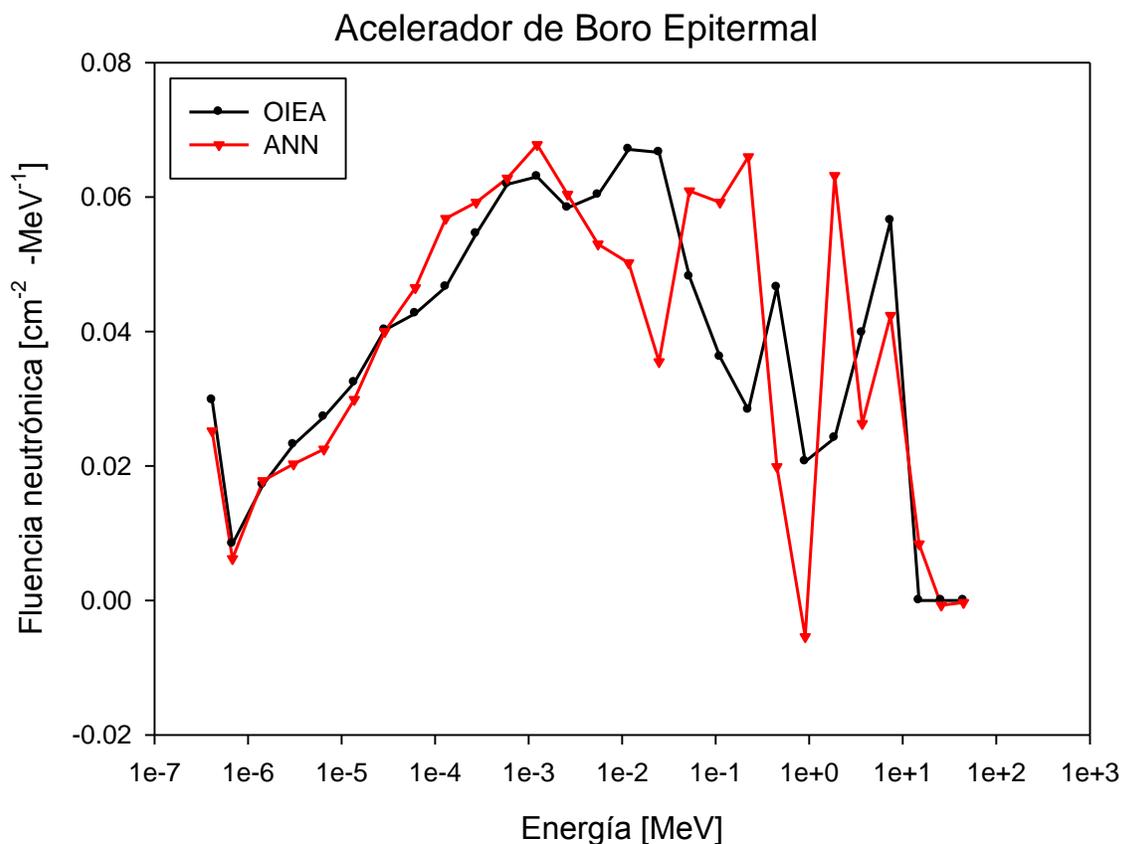


Figura 29. Espectros de neutrones de referencia y por ANN de Acelerador de Boro Epitermal.

Para la validación no se tomaron espectros de todos los grupos porque la gran mayoría son mono-energéticos y tienen muy bajo grado de robustez, así que aquí sólo se muestran los ejemplos más significativos. Para la prueba estadística  $\chi^2$  y ver que no hubiera diferencia entre espectros se tomaron: si tomamos como referencia la salida de la ANN y  $\alpha=0.05$  (nivel de confianza del 95%) y 6 grados de libertad, el valor crítico de  $\chi^2$  es de 12.592. Para ambos espectros, todos los valores son menores al valor crítico de  $\chi^2$ , lo que implica que las diferencias entre los espectros esperados del IAEA y los calculados mediante la ANN no son estadísticamente significativas.

## CONCLUSIONES

1. Para la obtención de las tasas de conteo de una fuente de neutrones es necesario obtener las mediciones de fondo, fuente y de la fuente con cono de sombra diseñado para cada esfera, ya que al omitir esta última medición y al hacer la corrección por fondo y cono de sombra, el sistema de deconvolución no arrojará buenos resultados. Lo más relevante es que para simular las cuentas en la ANN, las mediciones con el BSS se tienen que hacer un mayor número de veces para cada caso, esto porque el error para detección de neutrones es amplio y la ANN está entrenada con datos de espectros de neutrones calibrados, obtenidos de aceleradores, reactores nucleares, de fuentes isotópicas de referencia y funciones matemáticas.
2. Para el diseño de una ANN óptima es necesario usar una base de datos amplia y lo más exacta posible, para el espectro de neutrones genera mayor problema porque estos no se pueden medir directamente y algunos espectros son robustos por tanto se deben promediar con más ANN, en el caso de las dosis la matriz debe ser menos compleja porque debe ser más pequeña.
3. El diseño de la arquitectura de la ANN elegida es de 25:25:27, red tipo *feedforward*, pero una red con más capas y neuronas tarda mucho tiempo y para hardware simple las ANNs presentan restricciones de memoria y procesamiento. El algoritmo de entrenamiento *trainscg* funciona bien para este problema ya que es el único que soporta la GPU.
4. Al probar una ANN con GPU muestra ser buena alternativa para el procesamiento ya que se usa el supercómputo del ordenador y así la CPU no se ralentiza al hacer otros procesos, cabe mencionar que esto es para una gran cantidad de datos, otra ventaja es que cualquier ordenador se

puede acondicionar con supercómputo y no serían necesarias las costosas computadoras.

5. La combinación de arquitecturas de hardware arrojó un desempeño bajo porque al estar actualizando los pesos de la ANN y comparando el proceso con CPU y GPU se hace lento el sistema ya que son arquitecturas diferentes. El posible cuello de botella en el procesamiento de los datos cuando se utiliza CPU y GPU en una computadora convencional, es la velocidad de transmisión en el ducto PCI-E.

## REFERENCIAS

- AGOSTEO, S., BEDOGNI, R., CARESANA, M., CHARITONIDIS, N., CHITI, M., ESPOSITO, A., FERRARINI, M., SEVERINO, C. & SILARI, M. 2012. Characterization of extended range Bonner Sphere Spectrometers in the CERF high-energy broad neutron field at CERN. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 694, 55-68.
- ARYA, A. P. 1966. *Fundamentals of nuclear physics*, Boston, Allyn and Bacon.
- AVDIC, S., POZZI, S. A. & PROTOPOPESCU, V. 2006. Detector response unfolding using artificial neural networks. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 565, 742-752.
- AZA, E., DINAR, N., MANESSI, G. P. & SILARI, M. 2016. A Bonner Sphere Spectrometer for pulsed fields. *Radiation Protection Dosimetry*, 168, 149-153.
- BASHEER, I. A. & HAJMEER, M. 2000. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43, 3-31.
- BIRATTARI, C., DIMOVASIL, E., MITAROFF, A. & SILARI, M. 2010. A Bonner Sphere Spectrometer with extended response matrix. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 620, 260-269.
- BRAGA, C. C. & DIAS, M. S. 2002. Application of Neural Networks for unfolding neutron spectra measured by means of Bonner Spheres. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476, 252-255.
- BRAMBLETT, R. L. & BONNER, T. W. 1960. Neutron evaporation spectra from (p, n) reactions. *Nuclear Physics*, 20, 395-407.
- BRAMBLETT, R. L., EWING, R. I. & BONNER, T. W. 1960. A new type of neutron spectrometer. *Nuclear Instruments and Methods*, 9, 1-12.
- BRETTE, R. & GOODMAN, D. F. M. 2012. Simulating spiking neural networks on GPU. *Network: Computation in Neural Systems*, 23, 167-182.
- BURGETT, E. A., HERTEL, N. E. & HOWELL, R. M. 2009. Energy Response and Angular Dependence of a Bonner Sphere Extension. *IEEE Transactions on Nuclear Science*, 56, 1325-1328.
- CAI, X., LAI, G. & LIN, X. 2013. Forecasting large scale conditional volatility and covariance using neural network on GPU. *Journal of Supercomputing*, 63, 490-507.
- CEMBER, H. & JOHNSON, T. E. 2009. *Introduction to health physics*, New York, McGraw-Hill Medical.
- CRUZATE, J. A., CARELLI, J. L. & GREGORI, B. N. 2009. *Bonner Sphere Spectrometer : A CONRAD Project Intercomparison*, International Atomic Energy Agency (IAEA), IAEA.
- CHOW, T. W. S. & CHO, S.-Y. 2007. *Neural networks and computing : learning algorithms and applications*, London; Singapore; Hackensack, NJ, Imperial College Press ; Distributed by World Scientific.

- DECKER, A. W., MCHALE, S. R., SHANNON, M. P., CLINTON, J. A. & MCCLORY, J. W. 2015. Novel Bonner Sphere Spectrometer Response Functions Using MCNP6. *IEEE Transactions on Nuclear Science*, 62, 1689-1694.
- DUBEAU, J., HAKMANA WITHARANA, S. S., ATANACKOVIC, J., YONKEU, A. & ARCHAMBAULT, J. P. 2011. A neutron spectrometer using nested moderators. *Radiation Protection Dosimetry*, 150, No. 2, 217-222.
- DUKE, M. J. M., HALLIN, A. L., KRAUSS, C. B., MEKARSKI, P. & SIBLEY, L. 2016. A precise method to determine the activity of a weak neutron source using a germanium detector. *Applied Radiation and Isotopes*, 116, 51-56.
- FIDJELAND, A. K., GAMEZ, D., SHANAHAN, M. P. & LAZDINS, E. 2013. Three Tools for the Real-Time Simulation of Embodied Spiking Neural Networks Using GPUs. *Neuroinformatics*, 11, 267-290.
- GALLEGO, E., LORENTE, A., VEGA-CARRILLO, H. R. & INTERNATIONAL ORGANIZATION FOR, S. 2004. Characteristics of the neutron field of the facility at DIN-UPM. *Radiation Protection Dosimetry*, 110, 73-79.
- GUZMAN-GARCIA, K. A., MENDEZ-VILLAFANE, R. & VEGA-CARRILLO, H. R. 2015. Neutron field characteristics of Ciemat's Neutron Standards Laboratory. *Applied Radiation and Isotopes*, 100, 84-90.
- HAGAN, M. T., DEMUTH, H. B. & BEALE, M. H. 2016. *Neural network design*, Boulder, CO, Distributed by Campus Pub. Service, University of Colorado Bookstore.
- HAYKIN, S. S. 2007. *Neural networks : a comprehensive foundation*, Harlow, Prentice Hall.
- HERNANDEZ-DAVILA, V. M., SOTO-BERNAL, T. G. & VEGA-CARRILLO, H. R. 2014. Determination of neutron fluence-to-dose conversion coefficients by means of artificial neural networks. *Applied Radiation and Isotopes*, 83, Part C, 249-251.
- HO, T.-Y., LAM, P.-M. & LEUNG, C.-S. 2008. Parallelization of cellular neural networks on GPU. *Pattern Recognition*, 41, 2684-2692.
- HOWELL, R. M., BURGETT, E. A., WIEGEL, B. & HERTEL, N. E. 2010. Calibration of a Bonner sphere extension (BSE) for high-energy neutron spectrometry. *Radiation measurements*, 45, 1233-1237.
- HU, Y. H. & HWANG, J.-N. 2002. *Handbook of neural network signal processing*, Boca Raton, CRC Press.
- IAEA 2001. Compendium of Neutron Spectra and Detector Responses for Radiation Protection Purposes. *Supplement to Technical Report Series No. 318. Technical Reports Series 403*. Vienna: International Atomic Energy Agency.
- IZAURIETA, F. & SAAVEDRA, C. 1999. Redes neuronales artificiales. *Revista Charlas de Física*, 16, 1-15.
- KARDAN, M. R., KOOHI-FAYEGH, R., SETAYESHI, S. & GHIASSI-NEJAD, M. 2004. Fast neutron spectra determination by threshold activation detectors using neural networks. *Radiation Measurements*, 38, 185-191.
- KHABAZ, R. & MIRI, S. 2011. Development of a Bonner sphere spectrometer with emphasis on decreasing the contribution of scattering by using a new designed shadow cone. *Journal of Radioanalytical & Nuclear Chemistry*, 289, 789-794.
- KNOLL, G. F. 2000. *Radiation detection and measurement*, New York, Wiley.
- LAMARSH, J. R. & BARATTA, A. J. 2014. *Introduction to nuclear engineering*, Harlow, Pearson.

- MAGILL, J. & GALY, J. 2004. *Radioactivity - Radionuclides - Radiation : including the Universal Nuclide Chart on CD-ROM*, New York, NY, Springer-Verlag Berlin Heidelberg.
- MARES, V. & SCHRAUBE, H. 1994. Evaluation of the response matrix of a Bonner sphere spectrometer with LiI detector from thermal energy to 100 MeV. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 337, 461-473.
- MARTÍN DEL BRÍO, B. & SANZ MOLINA, A. 2006. *Redes neuronales y sistemas borrosos*, Madrid, Ra-Ma.
- MARTIN, J. E. 2006. *Physics for radiation protection a handbook*, Weinheim, Wiley-VCH-Verl.
- MARTÍNEZ-BLANCO, M. D. R., ORNELAS-VARGAS, G., SOLÍS-SÁNCHEZ, L. O., CASTAÑEDA-MIRANADA, R., VEGA-CARRILLO, H. R., CELAYA-PADILLA, J. M., GARZA-VELOZ, I., MARTÍNEZ-FIERRO, M. & ORTIZ-RODRÍGUEZ, J. M. 2016. A comparison of back propagation and Generalized Regression Neural Networks performance in neutron spectrometry. *Applied Radiation and Isotopes*, 117, 20-26.
- MATLAB. 2016a. *GUI de MATLAB* [Online]. Natick, MA: The MathWorks, Inc. Available: <https://es.mathworks.com/discovery/matlab-gui.html> [Accessed 8/12/ 2016].
- MATLAB. 2016b. *MATLAB: El lenguaje del cálculo técnico* [Online]. Natick, MA: The MathWorks, Inc. Available: <https://es.mathworks.com/products/matlab.html> [Accessed 8/12/ 2016].
- MATLAB. 2016c. *Neural Network Toolbox Documentation* [Online]. Natick, MA: The MathWorks, Inc. Available: <https://es.mathworks.com/help/nnet/> [Accessed 8/12/ 2016].
- MATLAB. 2016d. *Parallel Computing Toolbox Documentation* [Online]. Natick, MA: The MathWorks, Inc. Available: <https://es.mathworks.com/help/distcomp/> [Accessed 8/12/ 2016].
- MATLAB. 2016e. *Uso del cálculo GPU en MATLAB para GPUs compatibles con NVIDIA CUDA* [Online]. Natick, MA: The MathWorks, Inc. Available: <https://es.mathworks.com/discovery/matlab-gpu.html> [Accessed 8/12/ 2016].
- MAZROU, H., NEDJAR, A. & SEGUINI, T. 2016. Neutron spectrum measurements at a radial beam port of the NUR research reactor using a Bonner spheres spectrometer. *Applied Radiation and Isotopes*, 114, 63-70.
- MEI, S., HE, M. & SHEN, Z. 2014. Optimizing Hopfield Neural Network for Spectral Mixture Unmixing on GPU Platform. *IEEE Geoscience and Remote Sensing Letters*, 11, 818-822.
- MENDEZ-VILLAFANE, R., GUERRERO, J. E., EMBID, M., FERNANDEZ, R., GRANDIO, R., PEREZ-CEJUELA, P., MARQUEZ, J. L., ALVAREZ, F. & ORTEGO, P. 2014. Design and verification of the shielding around the new Neutron Standards Laboratory (LPN) at CIEMAT. *Radiation Protection Dosimetry*, 161, 393-7.
- MÉNDEZ-VILLAFANE, R., LOS ARCOS MERINO, J. M., DÍAZ, E. G. & LORENTE FILLOL, A. 2010. Determination of the emission rate of an Am-Be neutron source with a Bonner sphere spectrometer. *Radiation Measurements*, 45, 1271-1275.

- MOHAMMADI, N., HAKIMABAD, H. & MOTAVALLI, L. 2015. Neural network unfolding of neutron spectrum measured by gold foil-based Bonner sphere. *Journal of Radioanalytical & Nuclear Chemistry*, 303, 1687-1693.
- MØLLER, M. F. 1993. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, 6, 525-533.
- NAVEROS, F., LUQUE, N. R., GARRIDO, J. A., CARRILLO, R. R. & ROS, E. 2013. CPU-GPU hybrid platform for efficient spiking neural-network simulation. *BMC Neuroscience*, 14, 1-2.
- NVIDIA. 2016. *Developer Zone: CUDA Toolkit Documentation* [Online]. Santa Clara, CA: Nvidia Corporation. Available: <https://docs.nvidia.com/cuda/cuda-c-programming-guide/#abstract> [Accessed 10/12/ 2016].
- NVIDIA. 2017. *Aceleración computacional* [Online]. Santa Clara, CA: Nvidia Corporation. Available: <http://www.nvidia.es/object/gpu-computing-es.html> [Accessed 10/1/ 2017].
- OH, K.-S. & JUNG, K. 2004. GPU implementation of neural networks. *Pattern Recognition*, 37, 1311-1314.
- ORTIZ-RODRÍGUEZ, J. M., REYES ALFARO, A., REYES HARO, A., CERVANTES VIRAMONTES, J. M. & VEGA-CARRILLO, H. R. 2014. A neutron spectrum unfolding computer code based on artificial neural networks. *Radiation Physics and Chemistry*, 95, 428-431.
- PALLIPURAM, V., BHUIYAN, M. & SMITH, M. 2012. A comparative study of GPU programming models and architectures using neural networks. *Journal of Supercomputing*, 61, 673-718.
- SLAZYNSKI, L. & BOHTE, S. 2012. Streaming parallel GPU acceleration of large-scale filter-based spiking neural networks. *Network: Computation in Neural Systems*, 23, 183-211.
- SRIDHAR, A., VINCENZI, A., RUGGIERO, M. & ATIENZA, D. 2012. Neural Network-Based Thermal Simulation of Integrated Circuits on GPUs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits & Systems*, 31, 23-36.
- SWEEZY, J., HERTEL, N. & VEINOT, K. 2002. BUMS—Bonner sphere Unfolding Made Simple: an HTML based multisphere neutron spectrometer unfolding package. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476, 263-269.
- TABLADA, C. & TORRES, G. 2009. Redes Neuronales Artificiales. *Revista de Educación Matemática*, 24, 22-30.
- THIBAUT, J. & SENOCAK, I. 2012. Accelerating incompressible flow computations with a Pthreads-CUDA implementation on small-footprint multi-GPU platforms. *Journal of Supercomputing*, 59, 693-719.
- THOMAS, D. J. & ALEVRA, A. V. 2002. Bonner sphere spectrometers—a critical review. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476, 12-20.
- TSOULFANIDIS, N. & LANDSBERGER, S. 2011. *Measurement and detection of radiation*, Boca Raton, FL, CRC Press.
- TSUCHIDA, Y. & YOSHIOKA, M. 2013. Acceleration of Neural Network Learning by GPGPU. *Electronics & Communications in Japan*, 96, 59-66.
- VEGA-CARRILLO, H. R. 2002. TLD pairs, as thermal neutron detectors in neutron multisphere spectrometry. *Radiation Measurements*, 35, 251-254.

- VEGA-CARRILLO, H. R., HERNÁNDEZ-DÁVILA, V. M., MANZANARES-ACUÑA, E., GALLEGO, E., LORENTE, A. & IÑIGUEZ, M. P. 2007. Artificial neural networks technology for neutron spectrometry and dosimetry. *Radiation Protection Dosimetry*, 126, 408-412.
- VEGA-CARRILLO, H. R., ORTIZ-RODRIGUEZ, J. M., MARTINEZ-BLANCO, M. R. & HERNANDEZ-DAVILA, V. M. 2010. Artificial Neural Networks in Spectrometry and Neutron Dosimetry. *AIP Conference Proceedings*, 1310, 12-17.
- VINCENZI, A., SRIDHAR, A., RUGGIERO, M. & ATIENZA, D. 2011. Fast thermal simulation of 2D/3D integrated circuits exploiting neural networks and GPUs. *Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design*. Fukuoka, Japan: IEEE Press.
- VYLET, V. 2002. Response matrix of an extended Bonner sphere system. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476, 26-30.

# ANEXO A

## Certificado de calibración de la fuente de 3 Ci de <sup>241</sup>AmBe de la UPM.

588 24C

NUCLEAR MATERIALS AND EQUIPMENT CORPORATION  
 A Subsidiary of Atlantic Richfield Company  
 Quality Control Department

### NEUTRON SOURCE ASSEMBLY INSPECTION REPORT

Source Serial No. 930AM401

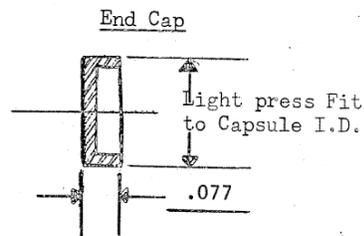
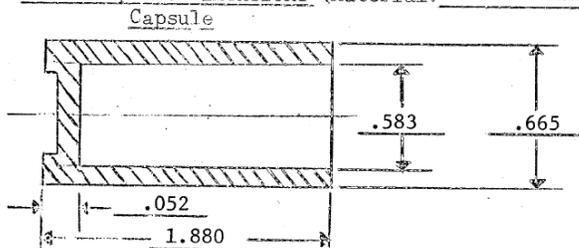
Source Type Am-Be

Customer Onuba, S. A.

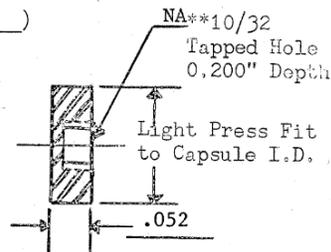
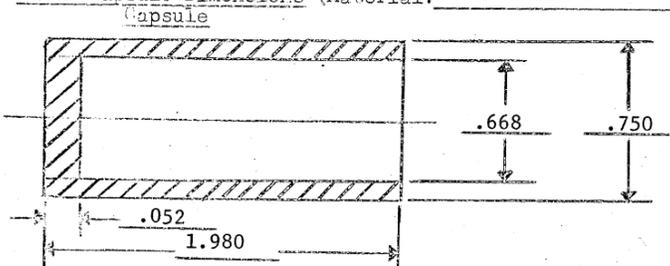
P. O. No. 7

NUMEC Contract No. 048-238

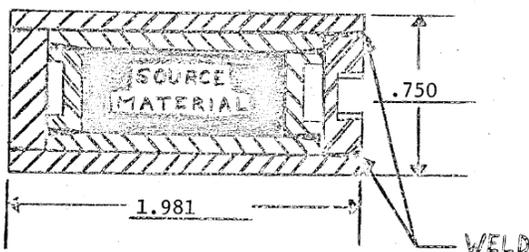
1. Inner Capsule Dimensions (Material: 304 L Stainless Steel)



2. Outer Capsule Dimensions (Material: 304L Stainless Steel)



3. Source Assembly Dimensions



4. Measured Neutron Emission From Source Assembly

$6.64 \times 10^6$  Neutrons/second

Calibrated against Numec Standard # 160A10

\*\*NOTE: (Source type shall determine whether tapped hole or solid end cap is used.)

Certified By: Joseph A. Dunlop Date Feb. 5, 1969  
 Quality Control

Americium Neutron Source Calibration and Certification

Serial Number - 930AM401

Date of Calibration - 1-15-69

Total Neutron Emission (Q) -  $6.64 \times 10^6$

Compared Against NUREC Standard - 160A10  
(Standard Calibrated by N.B.S.)

Stand Error -  $\pm 3\%$

Isotopic Analysis - The americium is "Isotopically Pure" as per supplier  
Union Carbide Nuclear Corporation, Oak Ridge,  
Tennessee

I certify that the material and/or services called for on Order  
Purchase Order 2 have been furnished in strict accordance  
with the requirements of the aforesaid order.

Signed: George Paul

Title: Foreman

Date: 1-16-69

## ANEXO B

Certificado de calibración de la fuente de  $^{252}\text{Cf}$  del CIEMAT.

### REPORT OF CALIBRATION

#### Neutron Source Strength Calibration Report

NIST Test Number: 282435R1 replaces report 282435  
Service ID: 44020C

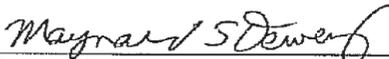
Calibration Performed for: QSA Global Inc  
Burlington, MA 01803

Neutron Source Description: Source Type:  $^{252}\text{Cf}$   
Serial Number: 76289B

Calibration Results:

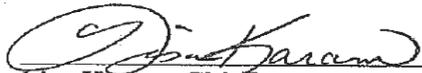
Calibrated neutron emission-rate:	$5.471 \times 10^8$ neutrons per second
Expanded uncertainty:	$\pm 2.6\%$ ( $2\sigma$ )
Calibration date:	May 20, 2012
NBS-1 emission-rate on date of calibration:	$1.230 \times 10^6$ neutrons per second

This calibration was performed by:

  
\_\_\_\_\_  
Maynard S. Dewey, Physicist  
Neutron Physics Group

For the Director of the National Institute of  
Standards and Technology:

  
\_\_\_\_\_  
Muhammad Arif, Leader  
Neutron Physics Group

  
\_\_\_\_\_  
Lisa Karam, Chief  
Radiation and Biomolecular Physics Division

**NIST**  
National Institute of  
Standards and Technology

Page 1 of 2

## Appendix for Cf Neutron Source Strength Calibration Reports

### Calibration Method

Neutron source strength measurements performed at NIST are accomplished by comparing the emission rate of the source being calibrated to that of the national primary standard neutron source, NBS-1, whose emission rate has been determined absolutely. The measurements of source emission rate are made by activating a circulating, aqueous solution of manganous sulfate, and continuously counting the induced  $^{56}\text{Mn}$  activity with a scintillation counter. During calibration, the neutron source is placed within a small Teflon cavity that is positioned at the center of the 1.3 m-diameter spherical bath; activity measurements are taken once the bath has reached saturation. The purpose of the cavity is to reduce the absorption of thermal neutrons by the source. Corrections to the measured source strength have been applied in order to account for the following effects: capture of fast neutrons by oxygen and sulfur in the bath, capture of fast and thermal neutrons by fluorine in the Teflon source holder, neutron escape from the bath, and thermal neutron absorption in the source. Typical values for these corrections are:

Fast neutron capture by oxygen and sulfur:	0.624 %
Fast and thermal neutron capture by fluorine:	0.170 %
Neutron escape from the bath:	0.045 %
Thermal neutron absorption in the source:	0.186 %

### Uncertainties

The expanded uncertainty consists of components evaluated by statistical means (the so-called Type A uncertainties) and components determined on the basis of alternative techniques (the so-called Type B uncertainties). The Type-A and Type-B uncertainty components relevant to this calibration are identified below.

Type A uncertainties: Count-rate associated with NBS-1 (typically < 0.5 %)  
Count-rate associated with the calibrated source (typically < 0.5 %)

Type B uncertainties: Uncertainty associated with the applied corrections for both NBS-1 and the calibrated source ( $\pm 0.3$  %, and typically 0.25 to 1 %, respectively)  
NBS-1 emission-rate ( $\pm 0.85$  %)  
Detector calibration ( $\pm 0.85$  %)

The expanded uncertainty reported corresponds to the quadrature-sum of the stated uncertainty components multiplied by a coverage factor equal to two (2). The expanded uncertainty, therefore, represents a two-standard-deviation ( $2\sigma$ ) estimate of the overall uncertainty.

# ANEXO C

## Especificaciones de NVIDIA® GeForce® GTX 960 2GB.

**PNY.**

### NVIDIA® GeForce® GTX 960 2GB

XLR8 Edition by PNY



#### Pure Performance, Ultimate Experience

The GeForce® GTX™ 960 delivers incredible performance, power efficiency, and cutting-edge gaming technologies that only NVIDIA Maxwell™ technology can offer.

This is the perfect upgrade to advanced gaming, letting you dominate the competition with 60% faster performance and twice the power efficiency than previous-generation cards\*. Unleash its massive overclocking potential and push your performance even further. Plus, it features VXGI for realistic lighting, support for smooth, tear-free NVIDIA G-SYNC™ technology, and Dynamic Super Resolution that enables 4K-quality gaming on 1080p displays.

The GTX 960 even gives you great gaming to go with NVIDIA GameStream™ technology that lets you stream PC games to NVIDIA SHIELD™ devices.

\*Compared to GeForce GTX 660

#### KEY FEATURES

- NVIDIA Super Resolution technology
- NVIDIA® GameWorks™ technology
- NVIDIA GameStream™ technology
- NVIDIA G-Sync™-Ready
- NVIDIA SHIELD™-Ready
- Microsoft DirectX 12
- NVIDIA GPU Boost 2.0
- NVIDIA Adaptive Vertical Sync
- NVIDIA Surround™ technology
- Support for four concurrent displays including:
  - Dual Dual-link DVI, HDMI 2.0, Three DisplayPort 1.2
- NVIDIA SLI®-Ready
- NVIDIA CUDA® technology
- PCI Express 3.0 support
- OpenGL 4.4 support
- OpenCL™ support

#### Available Option



GTX 960 2GB  
Reference Cooler

#### SPECIFICATIONS

##### NVIDIA GeForce GTX 960

PNY Part Number	VCGGTX9602XPB
UPC Code	751492580227
Card Dimensions	9.5" x 4.376" Dual-Slot
GPU	GeForce® GTX 960
CUDA Cores	1024
Core Speed	1127 MHz
Boost Speed	1178 MHz
Memory Size	2048 MB GDDR5
Memory Interface	128-bit
Memory Bandwidth (GB/sec)	112
Texture Fill Rate (billion/sec)	72
Resolution	4096 x 2160 (Digital)
Power Input	1x 6-Pin
Bus Type	PCI-Express 3.0 x16

#### MINIMUM SYSTEM REQUIREMENTS

- PCI Express-compliant motherboard with one dual-width x16 graphics slot
- One 6-pin PCI Express supplementary power connectors
- Minimum 400W or greater system power supply<sup>2</sup>
- 300MB of available hard-disk space
- 2GB system memory (4GB or higher recommended)
- Microsoft Windows 8, Windows 7, Windows Vista
- DVD-ROM drive and Internet connection

<sup>1</sup> Support for HDMI includes GPU-accelerated Blu-ray 3D support (Blu-ray 3D playback requires the purchase of a compatible software player from CyberLink, ArcSoft, Corel, or Sonic), x.v.Color, HDMI Deep Color, and 7.1 digital surround sound. Upgrade your GPU to full 3D capability with NVIDIA 3DTV Play™ software, enabling 3D gaming, picture viewing, and 3D web video streaming. See [www.nvidia.com/3dtv](http://www.nvidia.com/3dtv) for more details.

<sup>2</sup> Minimum system power requirement based on a PC configured with an Intel Core i7 3.2GHz processor.

<sup>3</sup> Graphics Card driver is not included in the box. GeForce Experience will download the latest GeForce driver from the Internet after install.

PNY Technologies, Inc. 100 Jefferson Road, Parsippany, NJ 07054 | Tel 973-515-9700 | Fax 973-560-5590 | [www.PNY.com](http://www.PNY.com)

Features and specifications subject to change without notice. The PNY logo is a registered trademark of PNY Technologies, Inc. All other trademarks are the property of their respective owners. © 2015 PNY Technologies, Inc. All rights reserved.

Ver. 12.01.15

# ANEXO D

## Especificaciones de NVIDIA® GeForce® GT 720 2GB.

1/2/2017

EVGA - Product Specs - EVGA GeForce GT 720 2GB

**EVGA** EVGA GeForce GT 720 2GB

Part Number: **02G-P3-2724-KR**



Make your entire PC experience faster with the new NVIDIA GeForce GT 720 dedicated graphics card. Enjoy performance that's superior to integrated graphics in all your PC multimedia applications, including 2x faster web browsing, 5x faster video editing, and 8x faster photo editing.

Play modern games with advanced and reliable dedicated graphics. 70% faster performance than integrated graphics means a rich and smooth gaming experience. Plus, get the power to drive up to three HD displays with a single card.

GeForce GT means speed, stability, and industry-leading NVIDIA drivers. Tap into GeForce Experience to automatically get the latest drivers and optimize game settings with a single click.

Learn more [here](#)

### SPECIFICATIONS

- Base Clock: 797 MHz
- Memory Clock: 1800 MHz Effective
- CUDA Cores: 192
- Bus Type: PCI-E 2.0
- Memory Detail: 2048MB DDR3
- Memory Bit Width: 64 Bit
- Memory Speed: 1.43ns
- Memory Bandwidth: 14.4 GB/s

### DIMENSIONS

- Height: 2.709in - 68.8mm
- Length: 6.6in - 167.75mm
- Width: Dual Slot



### KEY FEATURES

- NVIDIA PhysX technology
- PCI Express 2.0 support
- NVIDIA Adaptive Vertical Sync
- Microsoft DirectX 12 API (feature level 11\_0) Support
- NVIDIA 3D Vision Ready\*\*\*
- NVIDIA CUDA Technology with OpenCL support
- OpenGL 4.4 Support
- OpenCL Support
- NVIDIA FXAA Technology
- NVIDIA PureVideo HD Technology

### RESOLUTION & REFRESH

- Max Monitors Supported: 3
- 240Hz Max Refresh Rate
- Max Analog: 2048x1536
- Max Digital: 2560x1600

### REQUIREMENTS

- 300 Watt or greater power supply with a minimum of 20 Amp on the +12 volt rail.\*\*\*\*
- PCI Express, PCI Express 2.0 or PCI Express 3.0 compliant motherboard with one graphics slot.
- Windows 10 32/64bit, Windows 8 32/64bit, Windows 7 32/64bit, Windows Vista 32/64bit, Windows XP 32/64bit

\*DVI-D = Digital Only. Please do not connect to "DVI to VGA" adapter.

\*\*Support for HDMI includes GPU-accelerated Blu-ray 3D support (Blu-ray 3D playback requires the purchase of a compatible software player from CyberLink, ArcSoft, Corel, or Sonic), x.v.Color, HDMI Deep Color, and 7.1 digital surround sound. Upgrade your GPU to full 3D capability with NVIDIA 3DTV Play software, enabling 3D gaming, picture viewing, and 3D web video streaming.

\*\*\*NVIDIA 3D Vision and Surround require 3D Vision glasses and 3D Vision-ready displays. See [www.nvidia.com/get3d](http://www.nvidia.com/get3d) for more information

\*\*\*\*Minimum system power requirement based on a PC configured with an Intel Core i7 3.2GHz processor.



**EVGA Precision X**  
Designed from the ground up to support new GPU technology, EVGA Precision X redefines what overclocking software should be.  
[www.evga.com/precision](http://www.evga.com/precision)



**EVGA OC Scanner X**  
Featuring a brand new look and layout, this popular application has been adapted to show all of the latest GeForce vitals.  
[www.evga.com/ocscanner](http://www.evga.com/ocscanner)



**MODS RIGS**  
1k EVGA Bucks Sponsorship for every 200 posts. Come show off your rig and join in on one of the biggest things happening at EVGA.  
[www.modsrigs.com](http://www.modsrigs.com)



**24/7 Technical Support**  
EVGA is here for you day or night to help answer any questions!  
[www.evga.com/support](http://www.evga.com/support)



**EVGA GAMING**  
If you live to game, this is the place for you! We have the best tournaments, prizes and game servers.  
[www.evga.com/gaming](http://www.evga.com/gaming)



**TEAMEVGA**  
Follow EVGA on your favorite Social Networking sites like Facebook, Twitter, Steam, and the EVGA Gaming Community.  
[www.evga.com/teamevga](http://www.evga.com/teamevga)



### PRODUCT WARRANTY

This product is covered under EVGA's 3 year warranty which covers parts and labor. Further warranty extension is available upon registration within 90 days of purchase. For more details please visit <http://www.evga.com/warranty/>



# CURRICULUM VITAE

## Datos personales



Nombre: **Oscar Ernesto Alonso Muñoz**

Domicilio: **C. Valentín Gómez Farías #204, Colonia Gonzales Ortega Segunda Sección, Zacatecas, Zacatecas. C.P. 98087**

Correo Electrónico: **iqneto@hotmail.com**

Teléfono: **4921490491**

Celular: **4921020835**

## Datos académicos

### **2014 -2016 Maestría**

Maestría en Ciencias Nucleares

Orientación en Instrumentación Nuclear

Unidad Académica de Estudios Nucleares

Universidad Autónoma de Zacatecas

Estancia de investigación en el Departamento de Ingeniería Nuclear de la Escuela Técnica Superior de Ingenieros Industriales de la Universidad Politécnica de Madrid de forma coordinada con el Laboratorio de Metrología de Radiaciones Ionizantes del Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas en Madrid, España., del 25 de abril al 5 de junio del 2016.

### **2008-2012 Licenciatura**

Ingeniería en Comunicaciones y Electrónica

Orientación en Mecatrónica

Cédula profesional: 9042329

Unidad Académica de Ingeniería Eléctrica

Universidad Autónoma de Zacatecas

### **2004-2007 Preparatoria**

Preparatoria Estatal Francisco Goytia

## **Cursos, seminarios y participación extracurriculares**

- *“Taller para el avance de los proyectos de investigación”*, 40 horas, impartido en el Marco Escuela Internacional de Verano UAZ SPAUAZ, del 17 de junio al 5 de julio del 2016.
- *“Matlab y el Toolbox de Redes Neuronales”*, 40 horas, impartido en el Marco Escuela Internacional de Verano UAZ SPAUAZ, del 17 de junio al 5 de julio del 2016.
- *“Seminario de Educación y Capacitación sobre Ciencia y Tecnología de Reactores Rápidos”*, organizado por la IAEA y el ININ del 29 de junio al 3 de julio del 2015.
- *“4° Foro para productores de Chile y 1° Festival del Chile 2014”*, organizado por el Comité Sistema Producto Chile Zacatecas, del 27 al 29 de noviembre del 2014.

- Visita a NASA Space Center Houston, Texas, el 5 de abril del 2012.
- “*Diplomado en Síntesis, Caracterización y Simulación de Materiales Avanzados*”, 30 horas, organizado por la UAIE UAZ, del 24 al 26 de febrero del 2010.

## **Distinciones**

- Mejor promedio de la Maestría en Ciencias Nucleares, generación 2014-2016, otorgado en el marco de la conmemoración del Día del Estudiante por la UAZ, el 23 de mayo del 2016.

## **Trabajos presentados en congresos**

- “Reconstrucción del espectro de neutrones usando una Red Neuronal Artificial en CPU y GPU”.  
XVI Simposio Internacional de Dosimetría de Estado Sólido (ISSSD) dirigido por la Sociedad Mexicana de Radiación y Dosimetría en Tuxtla Gutiérrez, Chiapas, del 24 al 28 de septiembre del 2016.
- “Reconstrucción de espectro de neutrones usando una Red Neuronal Artificial (RNA) en GPU”.  
17<sup>vo</sup> Seminario de Investigación de la Universidad Autónoma de Aguascalientes, del 25 al 27 de mayo del 2016.

## **Artículos publicados**

- Hernández-Dávila, V.M., Moreno-Muñoz A., Vega-Carrillo, H.R., Alonso-Muñoz, O.E., Ortiz-López, M.A. *Reconstrucción del espectro de neutrones usando una Red Neuronal Artificial en CPU y GPU*. XVI Simposio Internacional de Dosimetría de Estado Sólido (ISSSD). (En revisión).