

**PROYECTO INTEGRADOR DE LA CARRERA DE
INGENIERÍA NUCLEAR**

**INCORPORACIÓN DE TÉCNICAS DE MUESTREO
MEDIANTE HISTOGRAMAS MULTIDIMENSIONALES
AL CÓDIGO DE SIMULACIÓN DE FUENTES DE
MONTE CARLO KDSOURCE**

**Lucas Ezequiel Ovando
Estudiante**

Dr. Ariel Marquez
Director

Ing. Zoe Prieto
Co-director

Miembros del Jurado
Dr. Edmundo Lopasso
Mg. Norberto Schmidt

23 de Mayo de 2025

Departamento de Física de Reactores y Radiaciones

Instituto Balseiro
Universidad Nacional de Cuyo
Comisión Nacional de Energía Atómica
Argentina

A todos por igual

Índice de símbolos

Índice de contenidos

Índice de símbolos	v
Índice de contenidos	vii
Índice de figuras	ix
Índice de tablas	xi
Resumen	xiii
Abstract	xv
1. Introducción	1
1.1. Contexto y motivación	1
1.2. Trabajos relacionados	5
1.3. Aportes específicos de este trabajo	6
2. Herramientas utilizadas	7
2.1. OpenMC	7
2.2. KDSource	8
2.3. Formatos de listas de partículas: MCPL y HDF5	8
2.4. Formato XML	9
2.5. Lenguajes de programación: Python, C y C++	9
3. Metodología para la generación de fuentes Monte Carlo mediante histogramas multidimensionales	13
3.1. Introducción general al método	13
3.2. Definición del espacio de fases ($E-r-\Omega$)	13
3.3. Procesamiento del archivo de partículas original	14
3.3.1. Preprocesamiento del archivo de partículas	14
3.3.2. Utilización de histogramas macro y micro	14
3.3.3. Métodos de discretización aplicados a histogramas macro y micro	16
3.4. Remuestreo de partículas en simulaciones Monte Carlo subsecuentes . .	17

3.5. Implementación computacional	18
4. Aplicación: Canal de vacío rodeado por Agua	21
4.1. Descripción del modelo simulado en OpenMC	21
4.2. Procesamiento del trackfile	22
4.2.1. Metodología del análisis	22
4.2.2. Descripción del trackfile	23
4.2.3. Bineado uniforme de microgrupos y macrogrupos	25
4.2.4. Bineado uniforme de microgrupos y macrogrupos con bordes manuales	28
4.2.5. Bineado adaptativo de microgrupos y macrogrupos	32
4.2.6. Bineado adaptativo de microgrupos y macrogrupos optimizado .	35
4.3. Resultados de la simulación comparativa	36
4.3.1. Validación de tallies y aplicación de técnicas de reducción de varianza	36
4.3.2. Síntesis y conclusiones	36
5. Validación Experimental: Conducto Número 5 del Reactor RA-6	37
5.1. Descripción del caso experimental del RA-6	37
5.2. Optimización del resampleo para la fuente experimental	37
5.3. Resultados de simulación en OpenMC y comparación experimental . .	37
5.3.1. Distribución espacial del flujo en detectores	37
5.3.2. Comparación ^3He : simulación vs. experimento	37
5.4. Discusión sobre resultados y validación	37
A. Implementación Computacional del Método	39
A.1. Código detallado en Python/C++	39
A.2. Pseudocódigo y comentarios	39
A.3. Archivos de configuración XML	39
A.4. Flujo de trabajo práctico	39
Bibliografía	41
Publicaciones asociadas	43
Agradecimientos	45

Índice de figuras

1.1.	Representación esquemática del núcleo de un reactor tipo piletas con canal de extracción. Se observa el núcleo central, rodeado por agua y blindaje biológico, así como un canal de extracción de neutrones. El gradiente de color indica cualitativamente la distribución de la población neutrónica típica en simulaciones Monte Carlo desde el núcleo.	2
1.2.	Esquema ilustrativo del proceso de desacople geométrico en simulaciones Monte Carlo para un reactor de investigación tipo piletas. En la primera etapa se registra un archivo de partículas en la entrada del canal de extracción, el cual es posteriormente utilizado para generar una fuente distribucional que permite simular eficientemente el canal de forma independiente. El gradiente de colores representa la disminución de la población neutrónica.	4
2.1.	Diagrama de soporte de MCPL.	9
2.2.	Etapa de detección: registro de partículas en formato MCPL a partir de una simulación en OpenMC.	10
2.3.	Etapa de procesamiento: construcción de una fuente distribucional a partir de un archivo MCPL.	10
2.4.	Etapa de producción: uso del archivo XML para generar nuevas partículas de forma offline o on-the-fly.	11
3.1.	Representación del espacio de fases con las variables $(E, x, y, z, \theta, \phi)$, donde E es la energía, x , y y z son coordenadas espaciales y θ y ϕ son las variables direccionales.	14
3.2.	Esquema ilustrativo de la estructura jerárquica de macrogrupos y microgrupos formando un árbol. Se resalta claramente la jerarquía entre variables.	15
3.3.	Esquema ilustrativo del proceso secuencial de generación de partículas a partir de la distribución jerárquica guardada, resaltando el muestreo sucesivo mediante números pseudoaleatorios.	18

4.1. Distribuciones de x vs y para el <i>trackfile</i> registrado en la primera corrida del tubo de vacío.	24
4.2. Distribuciones de letargia μ para el <i>trackfile</i> registrado en la primera corrida del tubo de vacío, con escala logarítmica en el eje y . Se observa la presencia de dos poblaciones: una concentrada en $\mu = 1$ y letargia mínima, montada sobre una distribución continua de neutrones colisionados.	24
4.3. Distribuciones de letargia vs μ para el <i>trackfile</i> registrado en la primera corrida del tubo de vacío. Se observa la presencia de un conjunto de neutrones graficados como un pixel en $\mu = 1$ y letargia mínima, indicando la naturaleza discreta de la fuente original. A su vez se observa la presencia de un segundo conjunto de neutrones colisionados, que se distribuyen en el plano de forma continua, con un agrupamiento sobre letargia = 20, indicando la termalización de los neutrones.	25
4.4. Comparación de la distribución de letargia entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado. Se observa el efecto de la discretización uniforme en la distribución de letargia.	26
4.5. Comparación de la distribución de μ entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado. Se observa el efecto de la discretización uniforme en la distribución de μ	26
4.6. Distribuciones de x vs y para el <i>trackfile</i> remuestreado utilizando discretización uniforme tanto para macrogrupos como para microgrupos. .	27
4.7. Comparación de la distribución de letargia entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado. Se observa el efecto de separar manualmente los neutrones con letargia mínima.	29
4.8. Comparación de la distribución de μ entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado. Se observa el efecto de separar manualmente los neutrones con $\mu = 1$	30
4.9. Distribuciones de x vs y para el <i>trackfile</i> remuestreado utilizando bordes manuales para separar el canal de vacío explicitamente.	31
4.10. Comparación de la distribución de letargia entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado.	33
4.11. Comparación de la distribución de μ entre el <i>trackfile</i> original y el <i>trackfile</i> remuestreado.	33
4.12. Distribuciones de x vs y para el <i>trackfile</i> remuestreado.	34

Índice de tablas

3.1. Ejemplo ilustrativo de la estructura típica de un archivo de partículas. El archivo original suele contener cientos de miles de partículas.	15
4.1. Divergencia KL parcial y total para la configuración Equal / Equal. .	28
4.2. Divergencia KL parcial y total para la configuración Equal / Equal. .	32
4.3. Divergencia KL parcial y total para la configuración Equal / Equal. .	35

Resumen

Este es el resumen en castellano.

La tesis debe reflejar el trabajo desarrollado, mostrando la metodología utilizada, los resultados obtenidos y las conclusiones que pueden inferirse de dichos resultados.

Palabras clave: FORMATO DE TESIS, LINEAMIENTOS DE ESCRITURA, INS-TITUTO BALSEIRO

Abstract

This is the title in English:

The thesis must reflect the work of the student, including the chosen methodology, the results and the conclusions that those results allow us to draw.

Keywords: THESIS FORMAT, TEMPLATES, INSTITUTO BALSEIRO

Capítulo 1

Introducción

1.1. Contexto y motivación

Las simulaciones Monte Carlo son la herramienta estándar cuando la complejidad geométrica o la marcada anisotropía angular del flujo neutrónico dificultan la aplicación de métodos determinísticos. Sin embargo, cuando las partículas atraviesan blindajes o regiones altamente absorbentes, la estadística disponible en la zona de interés se reduce considerablemente, lo que incrementa la incertidumbre en magnitudes físicas clave como el flujo escalar o la dosis equivalente ambiental, e incluso puede imposibilitar su cálculo si no llega ninguna partícula.

La incertidumbre estadística de estos resultados, por lo general, decrece lentamente con el número de historias simuladas, incrementando el costo computacional de forma prohibitiva.

En este trabajo abordaremos la temática de la obtención de resultados a la salida de canales de extracción de neutrones en reactores de investigación tipo pileta. En estos sistemas, los neutrones generados en el núcleo deben atravesar el agua de la pileta antes de alcanzar la entrada del canal de extracción. Durante una simulación desde el núcleo, la mayoría de los neutrones permanecen dentro del mismo y aquellos que logran salir al agua son en gran medida absorbidos a medida que penetran en esta. En consecuencia, son escasos los neutrones que finalmente alcanzan la entrada del canal, haciendo que resulte computacionalmente costoso obtener estadística suficiente a la salida del canal, en el ambiente circundante o en dispositivos experimentales.

En la Figura 1.1 se presenta un esquema ilustrativo del núcleo de un reactor de investigación tipo pileta con un canal de extracción de neutrones. Este esquema consta de un núcleo central rodeado por agua y rodeado externamente por un blindaje biológico. El canal de extracción permite obtener un haz de neutrones destinado a usos posteriores en investigaciones experimentales. Además, se muestra esquemáticamente mediante un gradiente de color las regiones con mayor estadística neutrónica y cómo

esta disminuye drásticamente al avanzar por el agua y el blindaje. Este esquema ejemplifica una simulación típica de núcleo, ilustrando claramente las dificultades que se tienen para obtener estadística lejos del núcleo.

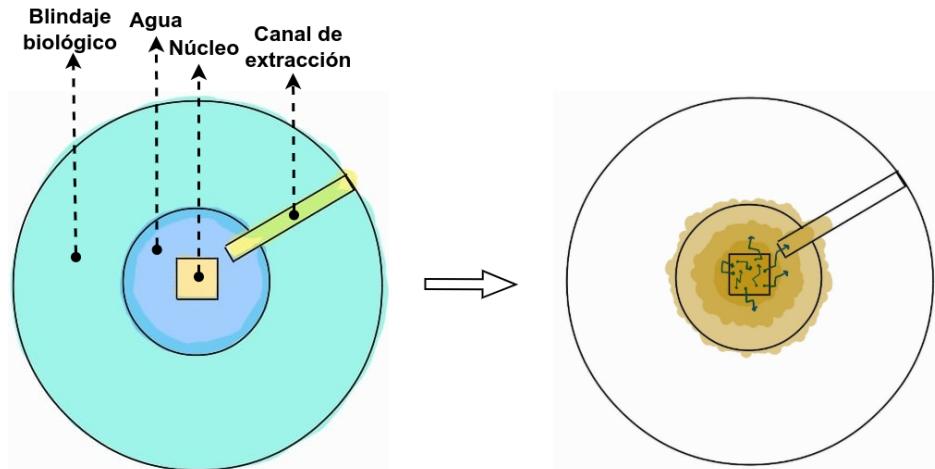


Figura 1.1: Representación esquemática del núcleo de un reactor tipo piletas con canal de extracción. Se observa el núcleo central, rodeado por agua y blindaje biológico, así como un canal de extracción de neutrones. El gradiente de color indica cualitativamente la distribución de la población neutrónica típica en simulaciones Monte Carlo desde el núcleo.

Para mitigar esta problemática, se suelen emplear técnicas de reducción de varianza tales como:

- **Separación geométrica del problema:** consiste en dividir la simulación en regiones contiguas, separando la fuente original de la región de interés mediante una superficie de interfaz. En esta superficie se registra la distribución de partículas en el espacio de fases y su peso estadístico. Esta información se emplea para construir una nueva fuente que se utiliza en una simulación posterior, permitiendo focalizar los recursos computacionales exclusivamente en la región de interés.
- **Ventanas de peso:** esta técnica define un rango aceptable de pesos estadísticos para las partículas en cada región del dominio. Aquellas con peso muy alto son divididas (*splitting*) y las de peso muy bajo pueden ser descartadas con cierta probabilidad, controlando así la varianza espacialmente.
- **Ruleta rusa:** se utiliza para descartar partículas con bajo peso estadístico, preservando el valor esperado del cálculo. Cada partícula tiene una probabilidad de ser eliminada o de continuar con su peso ajustado en consecuencia.
- **Sesgo por supervivencia:** en lugar de eliminar partículas cuando son absorbidas, se les reduce su peso estadístico de acuerdo a la probabilidad de absorción, permitiendo que las partículas sobrevivan y continúen su historia.

El método desarrollado en este trabajo se basa en la separación geométrica del problema en dos regiones: una que contiene el núcleo del reactor, y otra que abarca exclusivamente la región de interés. Esta técnica se implementa mediante la división del modelo simulado en dos regiones, delimitadas por superficies de acople ubicadas estratégicamente dentro de la geometría. En una primera etapa, se realiza una simulación completa del núcleo hasta una superficie intermedia, donde se registran en una lista las propiedades de las partículas que la atraviesan, incluyendo su energía, posición, dirección ($\mathbf{E}-\mathbf{r}-\Omega$), y su peso estadístico.

A partir de esta lista de partículas, se estima la distribución multidimensional en el espacio de fases utilizando histogramas multidimensionales. Esta estructura permite segmentar el conjunto original de datos mediante histogramas de baja resolución, llamados histogramas macro, que separan el espacio de fases en regiones donde las correlaciones entre variables se mantienen aproximadamente constantes. Sobre cada una de estas regiones, se construyen histogramas de mayor resolución —los histogramas micro— que aproximan la distribución de cada variable con mayor detalle. De este modo, se preservan tanto la forma general de la distribución como las correlaciones relevantes entre las variables registradas. A partir de esta estimación, es posible generar nuevas partículas que pertenezcan estadísticamente al mismo espacio de fases que la muestra original, obteniendo así una fuente sintética denominada fuente distribucional. Esta metodología se desarrolla en profundidad en el Capítulo 3, donde se describe el procedimiento implementado para su construcción.

En la segunda etapa, esta fuente distribucional se utiliza como fuente de entrada para una nueva simulación, que consiste únicamente en la región de interés, excluyendo el núcleo y sus alrededores. Al concentrar el esfuerzo computacional en esta zona, se incrementa la eficiencia, ya que las partículas simuladas poseen una mayor probabilidad de contribuir a las magnitudes físicas relevantes de estudio.

A su vez, además del método mencionado, en este trabajo también se hace uso del método de ventanas de peso como técnica complementaria de reducción de varianza.

En la Figura 1.2 se ejemplifica el proceso desarrollado para una simulación del núcleo de un reactor de investigación tipo piletta. Inicialmente, se realiza una simulación completa desde el núcleo, obteniendo una estadística adecuada en los alrededores del mismo, la cual se reduce considerablemente a medida que los neutrones penetran en el agua circundante. Este fenómeno se visualiza mediante un gradiente de colores. En esta simulación inicial, se define una superficie de registro en la entrada de un canal de extracción. Las partículas que atraviesan dicha superficie se registran en un archivo de partículas, el cual, mediante la metodología que se va a desarrollar en este trabajo, se procesa para definir una fuente distribucional. Posteriormente, esta fuente permite simular únicamente el canal de extracción, concentrando la estadística en dicha región sin necesidad de repetir la simulación completa desde el núcleo.

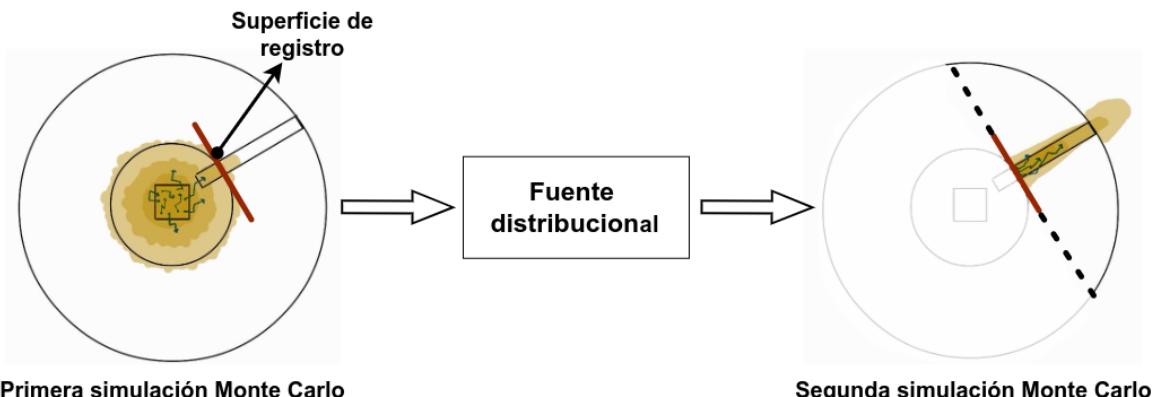


Figura 1.2: Esquema ilustrativo del proceso de desacople geométrico en simulaciones Monte Carlo para un reactor de investigación tipo piletas. En la primera etapa se registra un archivo de partículas en la entrada del canal de extracción, el cual es posteriormente utilizado para generar una fuente distribucional que permite simular eficientemente el canal de forma independiente. El gradiente de colores representa la disminución de la población neutrónica.

Este enfoque permite alcanzar una buena precisión en zonas alejadas o poco accesibles, con un número considerablemente menor de historias simuladas en comparación con una simulación completa desde la fuente original.

En resumen, el procedimiento consta de tres etapas fundamentales:

- **Detección:** Registro de las variables del espacio de fases y del peso estadístico de cada partícula que atraviesa una superficie intermedia de desacople. El resultado de este proceso es un archivo de partículas que contiene la información necesaria para caracterizar la distribución de la fuente sobre dicha superficie.
- **Estimación:** Procesamiento del archivo de partículas mediante la metodología propuesta en este trabajo, basada en histogramas multidimensionales. Este procedimiento permite aproximar la distribución en el espacio de fases, preservando las correlaciones entre variables, y genera como resultado una fuente distribucional.
- **Producción:** Utilización de la fuente distribucional estimada para generar nuevas partículas que pertenezcan al espacio de fases del archivo de partículas original. Estas partículas son empleadas en simulaciones posteriores, permitiendo modelar la región de interés de forma desacoplada de la fuente original.

El archivo original de partículas contiene inevitablemente regiones del espacio de fases con baja estadística o sin eventos registrados, producto de su carácter finito. Si se reutiliza directamente como fuente en una simulación posterior —por ejemplo, simulando varias veces las mismas partículas registradas— se corre el riesgo de reproducir el mismo ruido estadístico, afectando la calidad de los resultados.

Para mitigar este problema, se emplea un procedimiento de estimación basado en histogramas multidimensionales que permite densificar el espacio de fases. En este

enfoque, se utilizan histogramas para aproximar la distribución de variables, bajo el supuesto de que la probabilidad es constante dentro de cada bin. Esto habilita la generación de nuevas partículas en regiones que no estaban presentes explícitamente en el archivo original, completando así los vacíos estadísticos y asegurando una cobertura más uniforme del espacio de fases.

1.2. Trabajos relacionados

Los trabajos previos realizados en el Departamento de Física de Reactores y Radiaciones (DeFRRa) del Centro Atómico Bariloche (CAB) abordaron el modelado de fuentes distribucionales mediante el uso de histogramas multidimensionales [1–3]. Esta estrategia permitió capturar correlaciones entre energía, posición y dirección en geometrías planas rectangulares, siendo aplicada satisfactoriamente en estudios vinculados al reactor RA-10.

En continuidad con dichos desarrollos, la herramienta **KDSouce** —también desarrollada en el DeFRRa— introdujo técnicas más avanzadas basadas en *Kernel Density Estimation (KDE)*, en forma multivariante y adaptativa [4–7]. **KDSouce** permite ajustar automáticamente distribuciones continuas a partir de archivos de partículas previamente obtenidos y, a partir de ellas generar fuentes distribucionales, preservando las correlaciones del espacio de fases.

No obstante, el enfoque basado en *KDE* presenta una limitación: su carácter inherentemente suavizante puede dificultar la representación precisa de discontinuidades abruptas en el espacio de fases. Estas discontinuidades pueden originarse, por ejemplo, en zonas de cambio de material o en la coexistencia de poblaciones de partículas con comportamientos físicos muy distintos, como neutrones térmicos isotrópicos mezclados con neutrones rápidos colimados. En estos casos, la suavización puede degradar la calidad de la fuente generada, especialmente cuando se busca preservar estructuras finas o anisotropías marcadas.

El presente trabajo extiende estas líneas incorporando los enfoques basados en histogramas multidimensionales dentro del entorno ya establecido de **KDSouce**, proponiendo además un método de discretización adaptativa que automatiza la definición de las grillas de los histogramas macro y micro. A diferencia de los trabajos anteriores, este nuevo enfoque permite segmentaciones variables para diferentes subconjuntos del espacio de fases. Por ejemplo, el método puede asignar una discretización espacial más refinada para neutrones rápidos, donde suelen prevalecer anisotropías asociadas a haces colimados, y una discretización más homogénea para neutrones térmicos, cuya distribución tiende a ser más isotrópica. De esta forma, se logra preservar las correlaciones relevantes sin imponer una malla uniforme en todo el dominio, mejorando la capacidad de representación del método.

1.3. Aportes específicos de este trabajo

Este trabajo busca profundizar el desarrollo de `KDSource` mediante la incorporación de histogramas multidimensionales como una alternativa -o complemento- a la metodología *KDE* existente. Las contribuciones específicas son:

- Implementación de histogramas multidimensionales en `KDSource`, capaces de:
 - preservar las correlaciones entre variables ($\mathbf{E}-\mathbf{r}-\Omega$) para fuentes planas rectangulares;
 - representar adecuadamente discontinuidades o picos en todas las variables;
 - implementar un método de selección automática y adaptativa de bordes para los histogramas, tanto a nivel macro como micro, optimizado según la estadística disponible en cada subgrupo del espacio de fases.
- Integración optimizada del flujo de trabajo en `OpenMC` mediante:
 - generación *offline* de un archivo intermedio en formato XML contenido la fuente distribucional, que incluye la información de los histogramas multidimensionales;
 - desarrollo de un módulo en C que utilice eficientemente esta información para producir partículas individualmente;
 - implementación de un remuestreo *on-the-fly* integrado directamente en `OpenMC`, minimizando la ocupación de memoria al evitar la carga y gestión de archivos extensos de partículas.
- Validación del método en casos de complejidad creciente:
 - Un caso simplificado, consistente en un haz colimado ingresando en un paralelepípedo de agua atravesado por un canal de vacío, con fuentes definidas artificialmente para permitir una comparación con una simulación directa más extensa sin la aplicación del método desarrollado.
 - Un caso real, consistente en la propagación a través del conducto N°5 del reactor RA-6, utilizando un archivo de partículas proporcionada por el departamento de Física de Neutrones del CAB generada a través de una simulación del núcleo en `OpenMC`.

Capítulo 2

Herramientas utilizadas

Este capítulo tiene por objetivo presentar y describir las herramientas computacionales y teóricas utilizadas a lo largo de este trabajo. Específicamente, se abordará el programa `OpenMC`, la técnica estadística empleada denominada divergencia KL, los formatos de archivos usados (`MCPL` y `XML`), y los lenguajes de programación involucrados (`Python`, `C` y `C++`).

2.1. `OpenMC`

`OpenMC` es un programa de simulación Monte Carlo desarrollado inicialmente en el *Massachusetts Institute of Technology (MIT)* como *software* de código abierto [8]. Actualmente es mantenido por una activa comunidad internacional que contribuye continuamente a su desarrollo y expansión. Está especialmente orientado al cálculo del transporte de neutrones y fotones, permitiendo tanto simulaciones de criticidad como simulaciones de fuente fija. En este trabajo se emplean exclusivamente simulaciones de fuente fija.

Una de las ventajas de `OpenMC` es su flexibilidad en la obtención de resultados. El código permite registrar diferentes magnitudes físicas como flujos, dosis, corrientes, espectros energéticos, etc., además de ofrecer la posibilidad de registrar partículas que atraviesan superficies definidas por el usuario en un archivo de formato `h5` o `MCPL`. Estos archivos almacenan información de las variables de las partículas, permitiendo su posterior análisis o reutilización para generar nuevas simulaciones. Asimismo, `OpenMC` incorpora técnicas avanzadas de reducción de varianza, como las ventanas de peso (*weight windows*), que resultan fundamentales para reducir la incertidumbre estadística de los resultados en áreas de interés específico, técnica utilizada en este trabajo.

2.2. KDSource

`KDSource` es una herramienta computacional desarrollada en el Departamento de Física de Reactores y Radiaciones del Instituto Balseiro, cuyo objetivo principal es procesar archivos de partículas generados en simulaciones Monte Carlo para la construcción de nuevas fuentes distribucionales [9]. Su integración con `OpenMC` permite desacoplar geométricamente simulaciones complejas, facilitando significativamente el cálculo de transporte en geometrías difíciles o extensas.

El fundamento original de `KDSource` reside en la técnica **Kernel Density Estimation (KDE)**, una técnica estadística que permite estimar distribuciones continuas de variables a partir de muestras discretas, manteniendo la correlación existente entre ellas.

2.3. Formatos de listas de partículas: MCPL y HDF5

En simulaciones Monte Carlo, es común registrar partículas que atraviesan una superficie o ingresan a una región de interés, generando lo que se conoce como una lista de partículas. Estos archivos permiten capturar el estado de cada partícula —incluyendo su energía, posición, dirección y peso estadístico— al momento de cruzar una superficie.

`OpenMC` emplea por defecto el formato `HDF5` para almacenar estas listas [10]. Este formato binario jerárquico permite almacenar datos de manera eficiente, estructurada y accesible desde diversos lenguajes de programación. Sin embargo, su estructura está adaptada específicamente al ecosistema de `OpenMC`, dificultando su reutilización directa en otros códigos de transporte.

Con el objetivo de facilitar la interoperabilidad entre distintos códigos Monte Carlo, existe el formato **MCPL (Monte Carlo Particle List)** [11]. Este estándar de listas de partículas permite almacenar listas generadas en simulaciones de forma compacta y eficiente, preservando información esencial como la energía, posición, dirección y peso de cada partícula. A diferencia de formatos específicos como `HDF5`, `MCPL` fue concebido como una interfaz entre diferentes entornos Monte Carlo.

La Figura 2.1 muestra los diversos códigos que pueden producir o consumir archivos en formato `MCPL`. En este proyecto, su adopción permite vincular eficientemente los archivos de partículas generados por `OpenMC` con la herramienta `KDSource`.

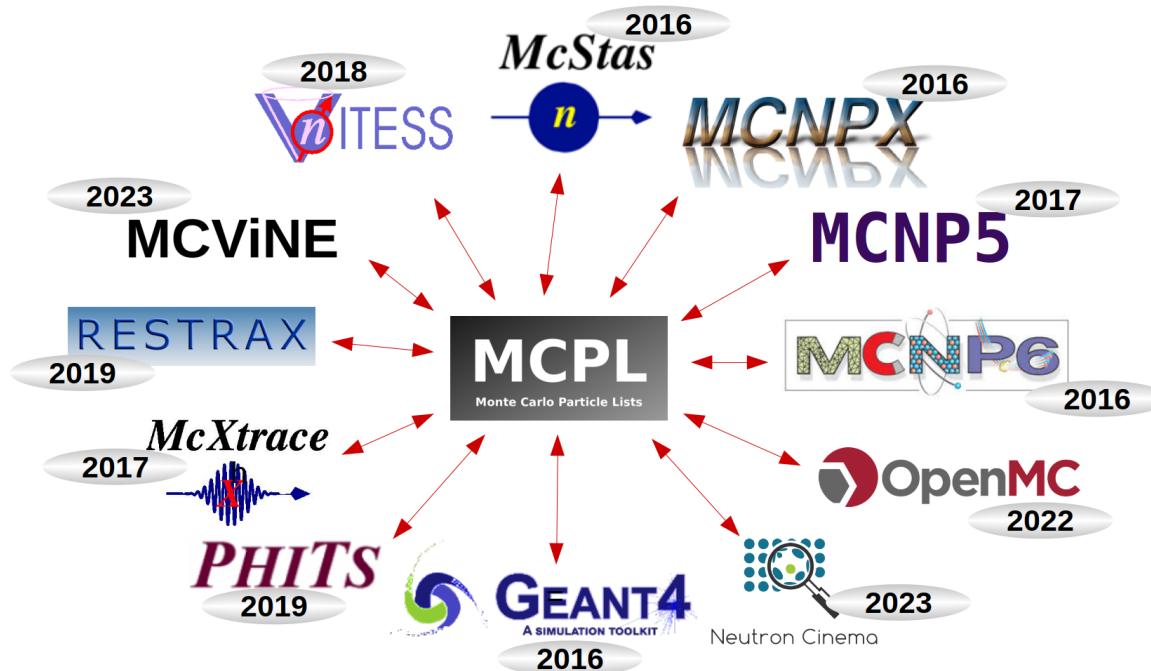


Figura 2.1: Diagrama de soporte de MCPL.

2.4. Formato XML

El formato **XML** (*Extensible Markup Language*) es un lenguaje utilizado para describir y almacenar información estructurada de manera jerárquica, basándose en una organización de datos tipo árbol [12]. Una de sus principales ventajas es que está diseñado para ser tanto legible por máquinas como por humanos, lo que facilita su edición, inspección y validación manual durante el desarrollo. Debido a estas características, resulta particularmente adecuado para guardar las distribuciones estimadas mediante histogramas multidimensionales. A su vez, OpenMC emplea archivos XML para almacenar la configuración completa de sus simulaciones (geometría, materiales, configuración de *tallies*, etc.), por lo que la elección de este formato contribuye a una integración directa y eficiente entre los componentes desarrollados.

2.5. Lenguajes de programación: Python, C y C++

A lo largo del desarrollo del proyecto, se utilizaron principalmente tres lenguajes de programación:

- **Python**: lenguaje de alto nivel especialmente adecuado para interfaces de usuario, análisis exploratorio de datos y configuración de simulaciones debido a su simplicidad, claridad y flexibilidad. KDSource hace uso de Python para la preparación

y procesamiento de datos previos al remuestreo Monte Carlo. Tanto `KDSouce` como `OpenMC` disponen de una interfaz en Python.

- **C:** lenguaje de programación de bajo nivel conocido por su eficiencia computacional, velocidad y control preciso sobre la gestión de memoria. Se utilizó en este trabajo para desarrollar módulos específicos encargados del reuestreo eficiente de partículas, especialmente cuando se requieren grandes volúmenes de datos. La biblioteca de remuestreo de `KDSouce` está escrita en este lenguaje.
- **C++:** `OpenMC` está implementado en C++, combinando la eficiencia de lenguajes de bajo nivel con capacidades avanzadas de abstracción y organización orientada a objetos. Esta arquitectura permite la expansión modular del código y facilita su mantenimiento.

El desarrollo realizado en este trabajo implementa un flujo computacional para el uso de fuentes distribucionales en simulaciones Monte Carlo. Este flujo se compone de tres etapas principales: detección, procesamiento y producción, y aprovecha las capacidades de los lenguajes Python, C y C++.

Detección: se realiza mediante una simulación inicial en `OpenMC`, en la cual se registra un archivo de partículas sobre una superficie. Este archivo puede guardarse en formato HDF5, que es el formato nativo de `OpenMC`, y luego ser convertido al formato MCPL si se desea continuar con el procesamiento. Alternativamente, si `OpenMC` fue compilado con soporte para MCPL, puede generarse directamente en dicho formato. El archivo resultante se almacena en disco para su uso posterior.

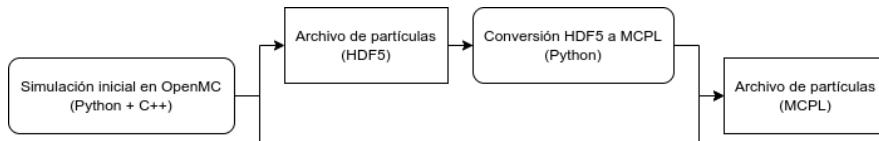


Figura 2.2: Etapa de detección: registro de partículas en formato MCPL a partir de una simulación en `OpenMC`.

Procesamiento: esta etapa se lleva a cabo en Python utilizando la implementación desarrollada para procesar mediante histogramas multidimensionales. A partir del archivo .MCPL, se construye una fuente distribucional que representa la densidad de probabilidad estimada en el espacio de fases. El resultado de esta etapa es un archivo XML, que contiene la parametrización de la fuente y se almacena en disco como entrada para la etapa de producción.

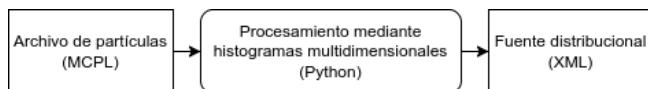


Figura 2.3: Etapa de procesamiento: construcción de una fuente distribucional a partir de un archivo MCPL.

Producción: esta etapa puede abordarse de dos maneras, implementada en código C. En la modalidad *offline*, el archivo XML es utilizado para generar una nueva lista de partículas en formato MCPL, que luego puede emplearse directamente como fuente en OpenMC si se cuenta con soporte para este formato. En caso contrario, puede convertirse nuevamente a HDF5. En la modalidad *on-the-fly*, se evita por completo el uso de listas intermedias: OpenMC accede directamente al archivo XML y realiza el remuestreo dinámicamente durante la simulación, gracias a una extensión *ad-hoc* incorporada en este trabajo.

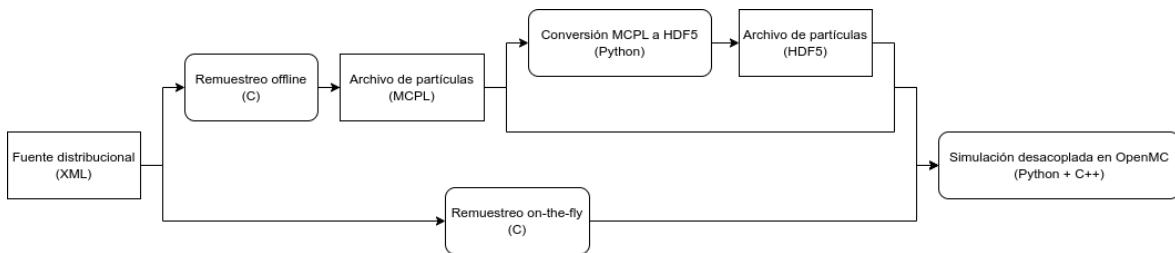


Figura 2.4: Etapa de producción: uso del archivo XML para generar nuevas partículas de forma offline o on-the-fly.

En los diagramas anteriores, los bloques con esquinas redondeadas representan procesos computacionales, mientras que los bloques rectangulares indican archivos que se registran o leen desde disco.

Capítulo 3

Metodología para la generación de fuentes Monte Carlo mediante histogramas multidimensionales

3.1. Introducción general al método

La metodología propuesta se basa en el procesamiento de archivos de partículas generados en simulaciones Monte Carlo previas, específicamente usando OpenMC, para definir fuentes distribucionales para simulaciones subsiguientes.

3.2. Definición del espacio de fases ($E-r-\Omega$)

Para aproximar correctamente las distribuciones y correlaciones del espacio de fases se consideraron seis variables para representar la energía, posición y dirección: tres coordenadas espaciales (x, y, z), dos variables direccionales definidas en coordenadas esféricas ($\mu = \cos(\theta), \phi$) y una variable energética (E o letargía $\ln(E_0/E)$). En la figura 3.1 se observa una representación de las variables del espacio de fases. En situaciones como la considerada en este trabajo, donde la fuente se registra sobre una superficie plana, es posible rotar el sistema de coordenadas de modo que dicha superficie resulte perpendicular al eje z . Por lo tanto la coordenada z permanece constante, permitiendo representar el espacio de fases con cinco variables.

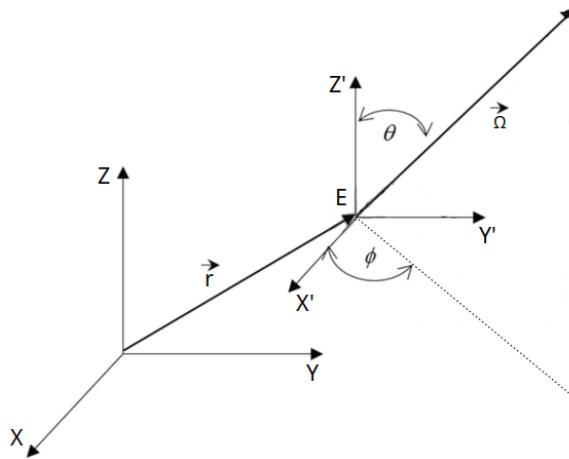


Figura 3.1: Representación del espacio de fases con las variables $(E, x, y, z, \theta, \phi)$, donde E es la energía, x , y y z son coordenadas espaciales y θ y ϕ son las variables direccionales.

3.3. Procesamiento del archivo de partículas original

3.3.1. Preprocesamiento del archivo de partículas

Inicialmente, las partículas provenientes de la simulación Monte Carlo original se filtran seleccionando únicamente aquellas que se propagan hacia la región de interés y se separan las variables relevantes mencionadas, además del peso estadístico (*weight*). Esta selección se realiza porque, en la siguiente etapa de simulación, la geometría considera un vacío en la región donde originalmente se encontraba la superficie de registro, permitiendo que únicamente las partículas que avanzan en dirección hacia la región de interés puedan ser simuladas. Las partículas que se dirigen en sentido opuesto no son relevantes para la simulación posterior, ya que en caso de que alguna partícula inicialmente dirigida hacia atrás fuese a regresar por retrodispersión, dicho comportamiento ya está estadísticamente incorporado en la lista original de partículas, producto de la simulación completa previa. En la Tabla 3.1 se presenta un ejemplo de la estructura típica de un archivo de partículas.

3.3.2. Utilización de histogramas macro y micro

La metodología desarrollada en este trabajo para aproximar distribuciones y correlaciones multidimensionales en el espacio de fases de partículas provenientes de simulaciones Monte Carlo se basa en un esquema combinado de histogramas macro y micro. Esta aproximación permite gestionar eficientemente grandes volúmenes de datos manteniendo una representación precisa de la información esencial del conjunto original de partículas.

Los histogramas macro son subdivisiones jerárquicas del espacio de fases, realizadas

Partícula N°	Letargía	x [cm]	y [cm]	μ	ϕ [rad]	Peso estadístico
1	4.15	0.23	-1.10	0.85	3.14	1.00
2	4.95	-0.75	0.40	0.65	1.57	1.00
3	5.05	1.10	0.70	0.45	0.78	1.00
4	5.30	-0.50	-0.90	0.60	2.35	1.00
5	4.85	0.85	-0.20	0.95	1.25	1.00
:	:	:	:	:	:	:
100000	5.00	0.10	0.55	0.70	0.95	1.00

Tabla 3.1: Ejemplo ilustrativo de la estructura típica de un archivo de partículas. El archivo original suele contener cientos de miles de partículas.

siguiendo un orden determinado por el usuario. Por ejemplo, un posible orden es la *letargía*, seguida por las coordenadas espaciales (*X*, *Y*), y posteriormente por la dirección (μ , ϕ). Este procedimiento inicia dividiendo el conjunto original de partículas en macrogrupos según la primera variable elegida (por ejemplo, *letargía*). Cada macrogrupo así generado es posteriormente subdividido en nuevos macrogrupos en función de la siguiente variable (por ejemplo, la coordenada *X*), repitiéndose este proceso de manera iterativa para cada variable subsiguiente. El resultado es una estructura jerárquica en forma de árbol, donde cada rama representa un subconjunto específico del archivo original de partículas, ya que incluye las cinco variables consideradas. Cada nodo en la rama corresponde a una variable dentro de dicho subconjunto. En la Figura 3.2 se ilustra este procedimiento de subdivisión jerárquica del espacio de fases.

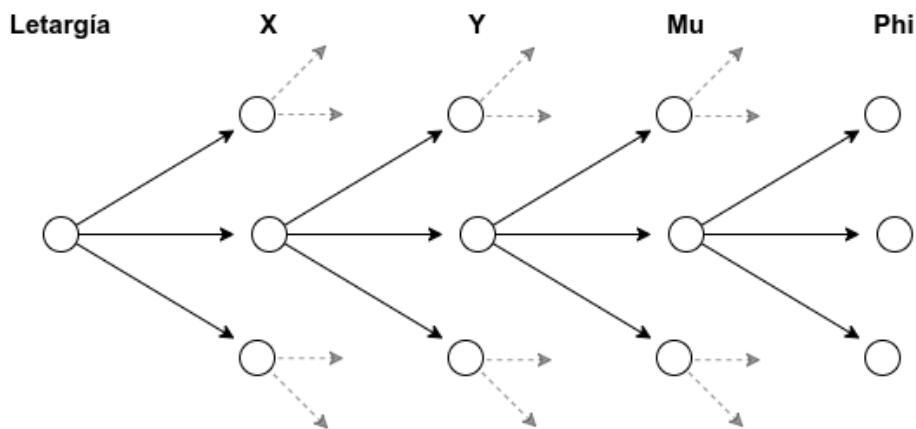


Figura 3.2: Esquema ilustrativo de la estructura jerárquica de macrogrupos y microgrupos formando un árbol. Se resalta claramente la jerarquía entre variables.

Este esquema tiene como objetivo capturar las correlaciones entre las variables del espacio de fases. Cada nodo o macrogrupo resultante comparte similitudes en todas las variables consideradas hasta ese nivel jerárquico. Por ejemplo, un macrogrupo particular tendrá partículas con distribuciones similares en *letargía*, coordenadas espaciales (*X*, *Y*), y dirección (μ), lo cual garantiza que la distribución restante en la última

variable (ϕ) refleje adecuadamente las correlaciones subyacentes en los datos.

Cabe destacar que, en la etapa de macrodiscretización, se utilizan histogramas con baja resolución debido al crecimiento exponencial en la cantidad total de grupos generados, dado que el número total de macrogrupos es producto del número de divisiones en cada variable. Un número excesivamente alto de divisiones podría conducir a subconjuntos con muy pocas partículas, comprometiendo la calidad estadística.

Una vez establecidos los macrogrupos, cada nodo es discretizado con histogramas micro, los cuales poseen una mayor resolución para representar detalladamente la distribución de cada variable dentro del subconjunto. Sin embargo, es crucial controlar esta resolución para evitar reproducir el ruido estadístico presente en los datos originales, especialmente cuando se cuenta con un número reducido de partículas en el subconjunto.

La combinación de histogramas macro y micro permite obtener una aproximación eficiente y precisa de la distribución multidimensional del conjunto original, conservando las correlaciones entre las variables del espacio de fases. Esta estructura de histogramas multidimensionales representa, en síntesis, la información esencial extraída del archivo original de partículas para ser utilizada en simulaciones posteriores.

3.3.3. Métodos de discretización aplicados a histogramas macro y micro

En este trabajo se desarrollaron dos métodos de discretización (bineado) aplicables tanto a los histogramas macro como a los micro. El primero es el *bineado uniforme*, que consiste en dividir la distribución de la variable en estudio utilizando bines equiespaciados. Este método tiene la ventaja de ser sencillo y rápido en su implementación, pero presenta la desventaja de no poder capturar adecuadamente cambios abruptos o picos aislados en las distribuciones.

El segundo método, denominado *bineado adaptativo*, consiste en realizar inicialmente un bineado uniforme con una cantidad moderada de bines especificada por el usuario. Luego, este bineado inicial se refina iterativamente agregando nuevos bines donde la distribución estimada difiere en mayor medida respecto a la distribución original. El criterio para añadir un nuevo bin se basa en la comparación de la función de distribución acumulada (FDC) del bineado actual con la FDC de la distribución original calculada con alta resolución. En cada iteración, se identifica el punto con la mayor diferencia absoluta entre ambas FDC, donde se coloca un borde de bin adicional.

Figura: Esquema gráfico del procedimiento de bineado adaptativo, mostrando las distribuciones estimadas con pocos bines, la distribución real y la diferencia absoluta utilizada para definir nuevos bines. **TO DO**

Este enfoque adaptativo permite asignar mayor resolución donde existe una mayor

cantidad de peso estadístico, y menor resolución donde el peso es escaso, logrando así un mejor seguimiento de las zonas críticas y un adecuado suavizado en regiones con poca estadística.

Además, debido a que la cantidad de partículas generalmente disminuye al profundizar en las ramas del árbol jerárquico, puede ser necesario utilizar una resolución decreciente en las discretizaciones, tanto para los histogramas macro como para los micro, lo cual también ha sido implementado en este trabajo.

Finalmente, en aquellos casos en que el usuario conozca previamente la existencia de cambios abruptos o picos específicos en la distribución, es posible informar manualmente al método sobre la ubicación de estos fenómenos, estableciendo bordes de bin específicamente en esos puntos para mejorar la precisión de la discretización.

3.4. Remuestreo de partículas en simulaciones Monte Carlo subsecuentes

El proceso de generación de partículas a partir de la información guardada en los histogramas multidimensionales implica:

1. Generar un número pseudoaleatorio entre 0 y 1.
2. Interpolar dicho número en la distribución acumulada normalizada de la variable raíz del árbol.
3. Avanzar secuencialmente por las ramas del árbol determinando valores de las variables subsecuentes hasta obtener un conjunto completo de variables del espacio de fases.

En la Figura 3.3 se ejemplifica el proceso de generación de partículas descrito previamente. En el caso particular de trabajar con cinco variables, se repetirá la generación de números pseudoaleatorios cinco veces consecutivas, obteniéndose así los cinco valores correspondientes de las variables. Cabe destacar que durante el remuestreo de la quinta variable no se generará un macrogrupo adicional, dado que esta es la última variable a muestrear y no existen más niveles en el árbol.

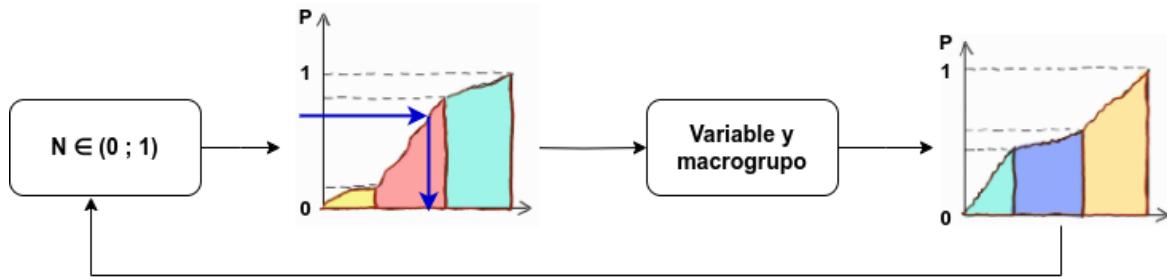


Figura 3.3: Esquema ilustrativo del proceso secuencial de generación de partículas a partir de la distribución jerárquica guardada, resaltando el muestreo sucesivo mediante números pseudoaleatorios.

A través de este proceso es posible remuestrear nuevas partículas para la siguiente etapa de la simulación. Existen dos formas de integrar esta funcionalidad con `OpenMC`:

- Una opción consiste en realizar, de forma *offline*, el muestreo de una cantidad predeterminada de partículas y guardar sus propiedades en un archivo de partículas. Este archivo puede ser luego utilizado por `OpenMC` mediante su opción de simulación desde el modo de fuente *FileSource* implementado en el programa.
- Alternativamente, se puede ejecutar `OpenMC` con una fuente del tipo `HistogramSource`, definida *ad hoc* y configurada mediante un archivo XML. En este caso, `OpenMC` accede directamente al árbol de histogramas durante la simulación y genera cada partícula *on-the-fly*, lo que reduce significativamente el uso de memoria y evita la necesidad de almacenar archivos de partículas intermedios voluminosos.

Esta funcionalidad fue incorporada en el contexto del presente trabajo, mediante la definición de una nueva clase de fuente en el código fuente de `OpenMC` en C++. Se desarrollaron las estructuras necesarias en C para efectuar el muestreo desde histogramas multidimensionales, y se integraron con la API de `OpenMC` en Python, permitiendo así que los histogramas puedan ser cargados y utilizados dentro del flujo habitual de simulación. El objetivo principal de esta implementación es facilitar el uso práctico de la herramienta.

3.5. Implementación computacional

La implementación computacional de la metodología descrita en este capítulo se encuentra desarrollada y documentada en el **Anexo A**. En particular, se presentan los códigos elaborados en Python, C y C++ que posibilitan la generación y manejo de los histogramas multidimensionales generados a partir del archivo original de partículas. Dichos histogramas se configuran mediante parámetros específicos definidos por el

usuario, tales como número y tipo de bins, así como el orden en que se procesan las variables del espacio de fases.

Asimismo, se documenta en detalle la integración realizada con `OpenMC`, incluyendo las modificaciones llevadas a cabo en su *API* de `Python` y en su código fuente en `C++`, para permitir la generación *on-the-fly* de partículas durante las simulaciones Monte Carlo subsecuentes.

Capítulo 4

Aplicación: Canal de vacío rodeado por Agua

En este capítulo se presenta una aplicación del método desarrollado para la generación de fuentes Monte Carlo mediante histogramas multidimensionales. El caso de estudio seleccionado consiste en un canal de vacío rodeado lateralmente por un bloque de agua liviana, simulando de forma aproximada una guía de neutrones. El objetivo del análisis es evaluar el desempeño del método propuesto en la generación de fuentes intermedias, con el fin de mejorar la estadística del flujo escalar en regiones específicas del modelo. (y conservar la corriente.)

Para ello, se realizaron varias etapas de simulación. Inicialmente, se efectuó una simulación para registrar un archivo de partículas (trackfile) en una superficie intermedia. Posteriormente, se llevó a cabo una simulación más extensa desde la entrada del sistema hasta otra superficie más alejada, con el propósito de obtener con precisión la distribución del flujo escalar a lo largo del canal, así como la corriente y el espectro neutrónico incidente sobre esta superficie final.

Finalmente, se procesó el trackfile grabado en la superficie intermedia para construir una fuente basada en histogramas multidimensionales. Esta fuente se utilizó en una tercera simulación iniciada directamente desde la posición intermedia, obteniendo así una estadística suficientemente convergida hasta la superficie más alejada. Esta última simulación permitió realizar una comparación directa con los resultados obtenidos en la simulación extensa desde la entrada, evaluando así la eficacia y precisión del método de generación de fuentes implementado.

4.1. Descripción del modelo simulado en OpenMC

El modelo utilizado en las simulaciones consiste en un canal interno de vacío rodeado lateralmente por un bloque de agua liviana. El sistema se diseñó como un paralelepípedo

de agua con dimensiones $15\text{ cm} \times 15\text{ cm} \times 100\text{ cm}$, que alberga en su interior un canal de vacío con sección transversal cuadrada de $3\text{ cm} \times 3\text{ cm}$, orientado en dirección del eje z .

La fuente de neutrones utilizada se definió sobre la cara de entrada del sistema (ubicada en $z = 0, \text{ cm}$) y consistió en neutrones monoenergéticos con energía de $E = 1 \text{ MeV}$ y colimados en dirección del eje z (con coseno direccional $\mu = \cos(\theta) = 1$). Esta configuración genera dos poblaciones de neutrones claramente diferenciadas: la primera, formada por neutrones que atraviesan el canal de vacío sin colisiones y mantienen tanto su energía inicial como su dirección; y la segunda, constituida por neutrones que interactúan con el moderador de agua, sufriendo dispersión angular y pérdida energética.

Para el análisis y posterior generación de la fuente, se registró un *trackfile* sobre una superficie intermedia situada en $z = 15 \text{ cm}$. Este archivo contiene información sobre ambas poblaciones mencionadas, es decir, neutrones no colisionados y neutrones dispersados.

Durante las simulaciones se emplearon tallies para determinar el flujo escalar neutrónico a lo largo del canal, tanto dentro del canal de vacío como en el moderador circundante.

Para optimizar la estadística en regiones con baja presencia de partículas, particularmente en el agua, se implementó la técnica de ventanas de peso adaptativo disponible en OpenMC. Este método actualiza dinámicamente los pesos de las partículas durante la simulación, permitiendo obtener mejores estadísticas con partículas de peso estadístico reducido.

4.2. Procesamiento del *trackfile*

4.2.1. Metodología del análisis

La estrategia general consiste en aplicar el método al *trackfile* registrado, utilizando distintos esquemas de configuración. En cada configuración se varían los parámetros: el tipo de bineado, el número de macrogrupos y microgrupos, y la posibilidad de indicar bordes de macrogrupo manualmente, y se analiza su impacto en la calidad de la reconstrucción. Para definir manualmente los bordes de los macrogrupos es necesario obtener información previa sobre la geometría o el comportamiento de las partículas que permita mejorar la separación de poblaciones con características distintas.

Los principales aspectos del análisis son:

- **Bineado uniforme vs. adaptativo:** se comparan histogramas de igual ancho y adaptativos. A su vez, para mejorar la representación del espacio de fase $\mathbf{E}-\mathbf{r}-\Omega$,

se analiza el efecto de incorporar bordes de macrogrupos definidos manualmente al caso de bineado uniforme. Esto permite ajustar la segmentación de los macrogrupos a la geometría del sistema o al comportamiento esperado de las partículas, mejorando la separación entre poblaciones y, por ende, la calidad del remuestreo.

- **Cantidad de macrogrupos:** se comparan esquemas uniformes (por ejemplo, [8, 8, 8, 8], [6, 6, 6, 6]) y esquemas crecientes o decrecientes (como [9, 8, 7, 6], [6, 7, 8, 9]). Esta elección impacta en la capacidad de representar correlaciones sin introducir exceso de ruido estadístico.
- **Cantidad de microgrupos:** se estudia el efecto de aumentar o disminuir la resolución dentro de cada macrogrupo.

Para cada configuración de entrada se comparan visualmente las distribuciones 1D y 2D obtenidas, analizando las distribuciones y sus errores relativos, como así también se analiza la divergencia KL obtenida para los casos 1D y 2D.

- **Distribuciones 1D:** se comparan las distribuciones de cada variable remuestreadas utilizando el método implementado contra las originales del *trackfile*.
- **Correlaciones 2D:** se representan visualmente mediante mapas de error relativo entre las matrices de correlación originales y las reconstruidas a través del remuestreo.
- **Métrica cuantitativa:** se utiliza la divergencia de Kullback-Leibler (KL) como estimador numérico de la diferencia entre las distribuciones, permitiendo comparar configuraciones de forma objetiva.

4.2.2. Descripción del trackfile

El *trackfile* analizado fue registrado en una primera simulación del tubo de vacío desde el comienzo de la geometría. Corresponde a partículas registradas en una superficie ubicada a 15cm del inicio de la geometría. La lista contiene tanto neutrones que atraviesan la sección del canal, como neutrones que viajan por el agua. Sin embargo, la mayor proporción de peso estadístico se encuentra dentro del tubo de vacío, lo cual se corrobora en la proyección $x-y$, donde se observa una distribución más intensa contenida en un cuadrado correspondiente a la sección del tubo. La figura 4.1 muestra la proyección de los neutrones en el plano $x-y$.

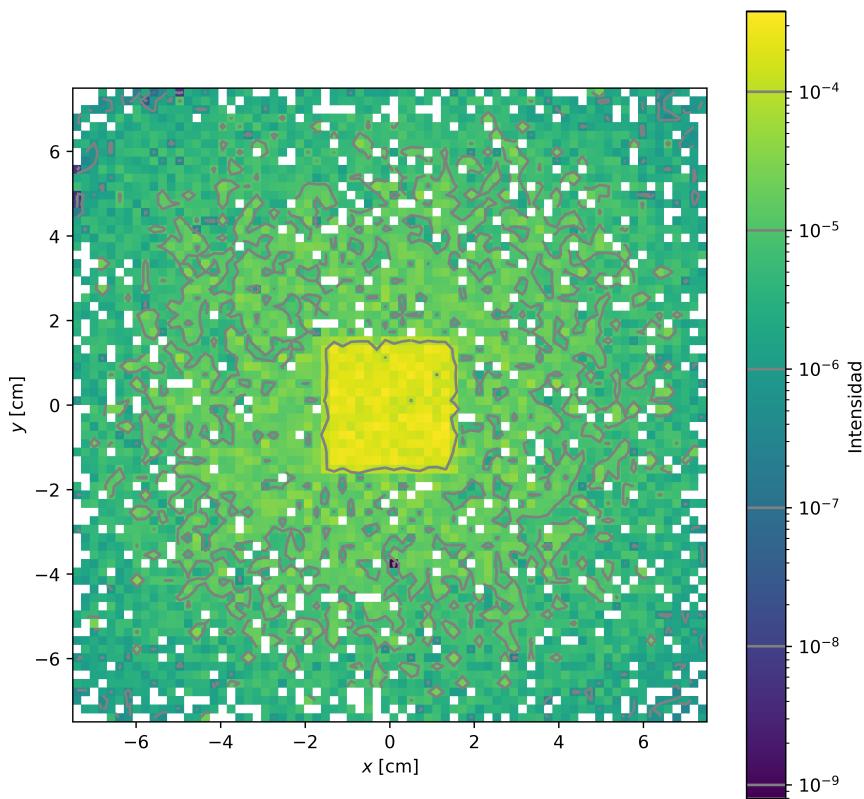


Figura 4.1: Distribuciones de x vs y para el *trackfile* registrado en la primera corrida del tubo de vacío.

Este conjunto contiene dos poblaciones de neutrones claramente diferenciadas. La primera corresponde a neutrones que no han sufrido colisiones: todos ellos poseen dirección $\mu = 1$ y una letargia mínima fija, sin dispersión alguna, lo cual indica que la fuente original fue configurada con estos valores de forma determinista. La segunda población incluye neutrones que han colisionado: en este caso, las distribuciones de μ y letargia son más amplias y continuas, reflejando la dispersión introducida por las interacciones. Este comportamiento se observa en la figura 4.2, donde se presentan las distribuciones de letargia y μ para el primer *trackfile*.

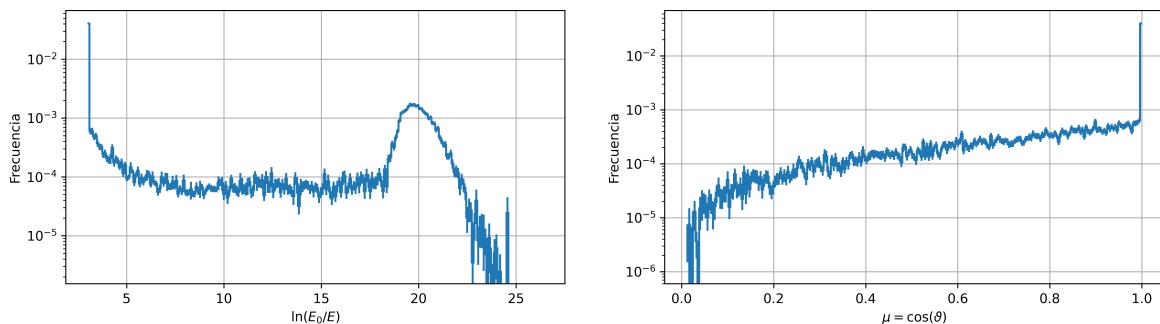


Figura 4.2: Distribuciones de letargia y μ para el *trackfile* registrado en la primera corrida del tubo de vacío, con escala logarítmica en el eje y. Se observa la presencia de dos poblaciones: una concentrada en $\mu = 1$ y letargia mínima, montada sobre una distribución continua de neutrones colisionados.

Esta doble estructura se evidencia particularmente en los gráficos 2D de letargia vs. μ , donde ambas poblaciones forman conjuntos visualmente separados (de hecho estamos hablando de dos conjuntos distintos. Repensar esto). Ver figura 4.3. Este fenómeno plantea un desafío para los métodos de muestreo, al requerir una representación precisa tanto de distribuciones concentradas tipo delta como de distribuciones extendidas y sus correlaciones.

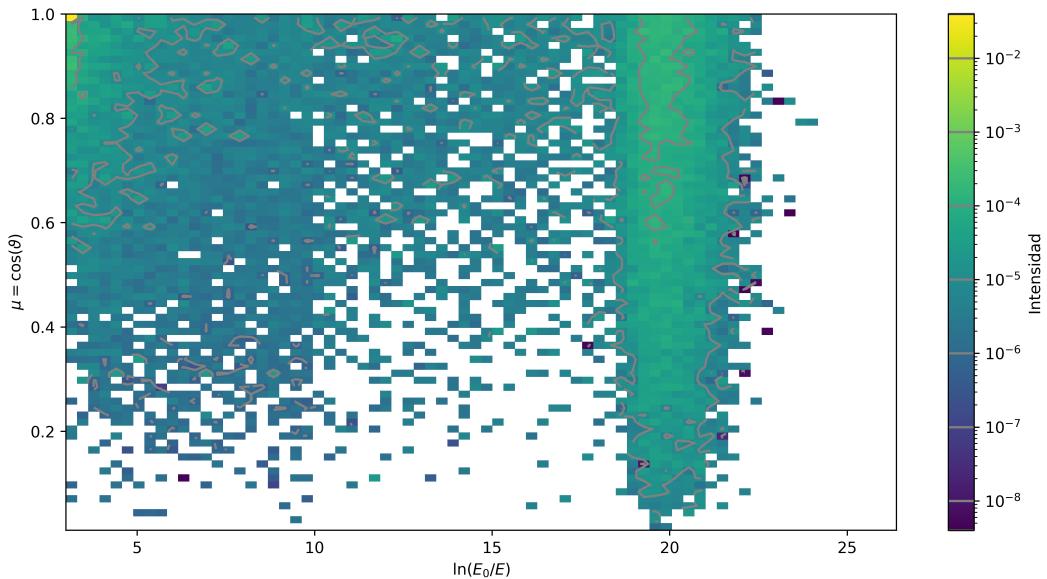


Figura 4.3: Distribuciones de letargia vs μ para el *trackfile* registrado en la primera corrida del tubo de vacío. Se observa la presencia de un conjunto de neutrones graficados como un pixel en $\mu = 1$ y letargia mínima, indicando la naturaleza discreta de la fuente original. A su vez se observa la presencia de un segundo conjunto de neutrones colisionados, que se distribuyen en el plano de forma continua, con un agrupamiento sobre letargia = 20, indicando la termalización de los neutrones.

A su vez se puede ver tanto en la figura 4.1 como en la figura 4.3 que aunque el bineado del plot 2D se haya hecho con 75 bins se observa claramente que existen muchos lugares del espacio de fase donde no hay neutrones. Esto es debido a que el trackfile original tiene un número limitado de neutrones. El objetivo de este método propuesto consiste en poder remuestrear el trackfile original para obtener un trackfile nuevo donde se pueda observar una mejor estadística en los lugares donde el trackfile original tiene pocos neutrones.

4.2.3. Bineado uniforme de microgrupos y macrogrupos

Se empleó una configuración con macrogrupos y microgrupos de ancho uniforme. Se seleccionaron $n_{macro} = 6$ y $n_{micro} = 50$ para cada variable, en el orden de variables: $[\ln(E_0/E), x, y, \mu, \phi]$. Esta configuración permite evaluar el desempeño base sin ningún tipo de adaptación.

Visualmente, se observa que la distribución tipo delta en letargia mínima y $\mu = 1$ es reemplazada por un rectángulo con el ancho del bin, perdiéndose el detalle de los picos originales. El efecto del resampleo mediante bins uniformes se puede ver en las figuras 4.4 y 4.5, donde se comparan las distribuciones de letargia y μ entre el trackfile original y el remuestreado.

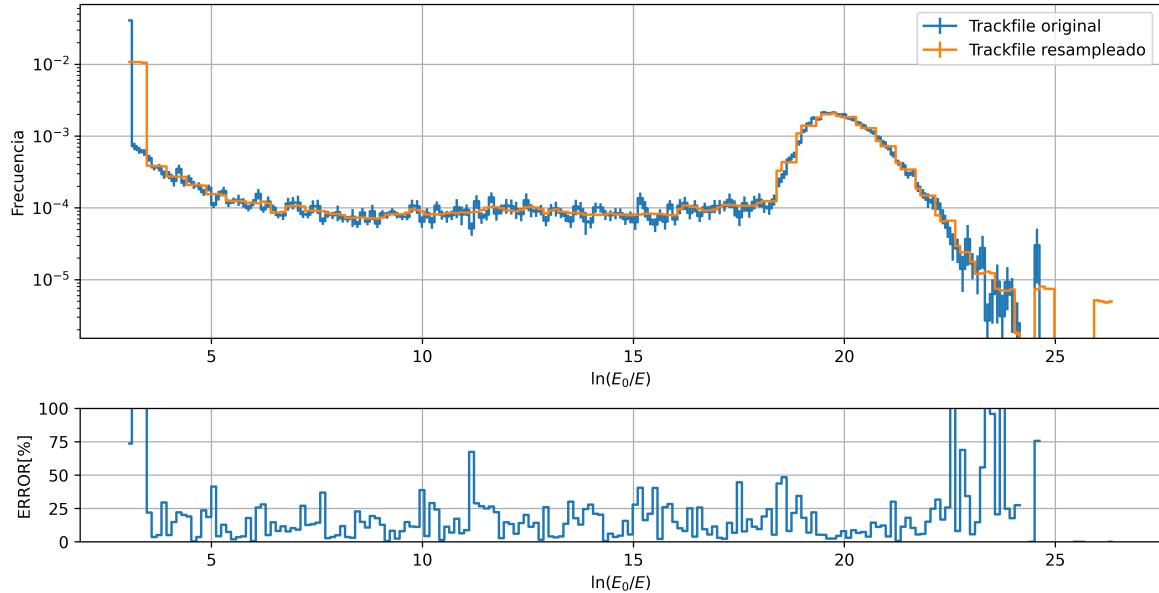


Figura 4.4: Comparación de la distribución de letargia entre el trackfile original y el trackfile remuestreado. Se observa el efecto de la discretización uniforme en la distribución de letargia.

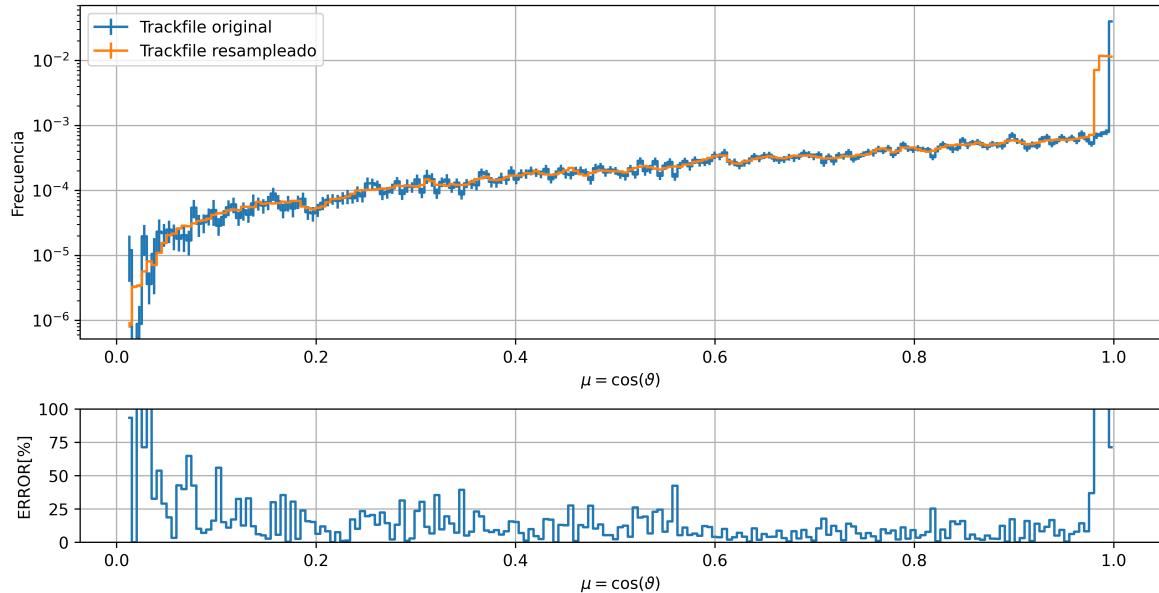


Figura 4.5: Comparación de la distribución de μ entre el trackfile original y el trackfile remuestreado. Se observa el efecto de la discretización uniforme en la distribución de μ .

Este efecto captado en las figuras 4.4 y 4.5 produce que los neutrones que originalmente viajaban por el vacío perfectamente colimados y monoenergeticos ahora se

descolimen y obtengan variacion en energia. Esto produce que los neutrones que viajaban cerca de la interfaz agua vacio ahora puedan ingresar al agua. Esto produce una perdida en la corriente por el vacio y una ganancia en la corriente por el agua.

Otro efecto que se observa es que la resolucion de los histogramas, al ser de bineado uniforme, no permite atribuir mas bins donde haya mas estadistica. Por lo tanto no se asignan mas bins a las deltas de mu y letargia, ni se asignan mas bins a la zona de termalizacion en la letargia.

Analizando el caso del ploteo 2D de x vs y se observa el efecto del bineado uniforme en los macrogrupos debido a que se observa que el resampling deforma la distribucion original. En la figura 4.1 se observa el canal de vacio y alrededor la estadistica del agua, de forma uniforme alrededor del canal. Sin embargo en la figura 4.6 se observa que se difumina parte de la estadistica del canal de vacio dentro del agua en forma de cruz. Esto se debe a que hay macrogrupos que intersectan el canal de vacio y el agua y produce que los neutrones que viajaban cerca de la interfaz agua vacio ahora puedan aparecer en el agua.

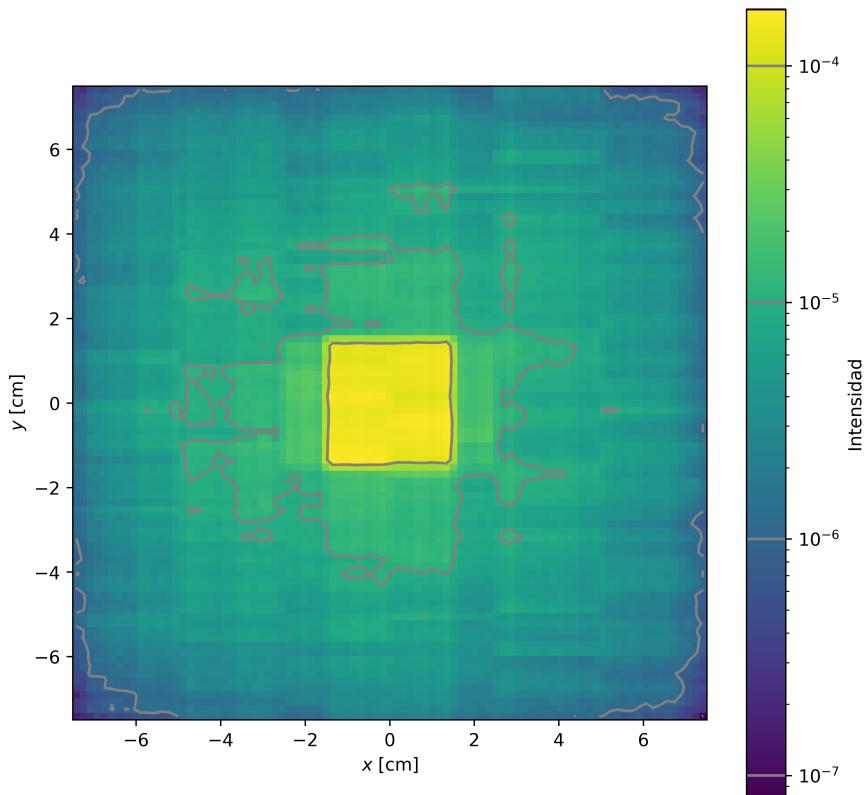


Figura 4.6: Distribuciones de x vs y para el *trackfile* remuestreado utilizando discretizacion uniforme tanto para macrogrupos como para microgrupos.

En la tabla 4.1 se presentan las divergencias KL 1D y 2D, tanto de forma parcial como total, para la configuración de bineado uniforme. Estos valores no indican informacion utilizable para decidir si el muestreo es aceptable o no. Sin embargo es util a

modo de comparacion contra otros bineados que posteriormente se van a mostrar.

Tabla 4.1: Divergencia KL parcial y total para la configuración Equal / Equal.

KL 1D		KL 2D	
Parámetro	Valor	Parámetros	Valor
$\ln(E_0/E)$	6.0372e-01	$\ln(E_0/E), x$	4.3450e+00
x	3.3379e-01	$\ln(E_0/E), y$	4.3407e+00
y	3.3882e-01	$\ln(E_0/E), \mu$	1.2565e+01
μ	1.3727e+01	$\ln(E_0/E), \phi$	4.4713e+00
ϕ	4.5827e-01	x, y	2.3899e+00
		x, μ	1.1431e+01
		x, ϕ	2.1703e+00
		y, μ	1.1432e+01
		y, ϕ	2.1283e+00
		μ, ϕ	1.1552e+01
Suma KL 1D 1.5462e+01		Suma KL 2D 6.6825e+01	

$$\text{KL total} = 8.2286\text{e}+01$$

4.2.4. Bineado uniforme de microgrupos y macrogrupos con bordes manuales

Para mitigar la pérdida de información debido a los problemas de resolucion del bineado uniforme, se incorporó la posibilidad de definir manualmente los bordes de los macrogrupos y microgrupos en las variables. Esto permitió aislar tanto la región correspondiente al canal de vacío, evitando que neutrones colimados se mezclen con partículas moderadas, como los neutrones de letargia mínima y $\mu = 1$.

Se definieron bordes manuales para las cuatro variables principales con el objetivo de aislar las regiones de mayor interés físico y evitar la mezcla entre poblaciones distintas. Los bordes se establecieron de la siguiente manera:

- **Letargia:** se fijó un borde en el valor mínimo de letargia + 1e-9 (correspondiente a los neutrones no colisionados) y el resto de los bordes se distribuyeron uniformemente en el rango restante.
- μ : se colocó un borde en $\mu = 1 - 1e - 9$ (dirección perfectamente colimada), separando así los neutrones no dispersados del resto; los demás bordes se ubicaron de forma uniforme en el intervalo restante.

- x e y : se asignaron bordes en las posiciones que delimitan el canal de vacío ($x = \pm 1,5 \text{ cm}$ y $y = \pm 1,5 \text{ cm}$), de modo que los macrogrupos coincidan con la geometría del canal y el agua circundante. El resto de los bordes se distribuyeron uniformemente fuera de esa región.

Esta segmentación permite preservar la física del sistema y mejorar la representación de las poblaciones diferenciadas en el proceso de remuestreo.

Al procesar el *trackfile* original con esta configuración, se observa que la distribución de letargia y μ se mantiene más cercana a la original, aunque aún se presenta un efecto debido a la falta de resolución donde hay más estadística. En la figura 4.7 se observa que la distribución remuestreada conserva la delta en letargia mínima. Sin embargo sería mejor obtener más bins en la zona de letargia mínima y en la zona de termalización. En la figura 4.8 se observa que la distribución remuestreada conserva la delta en $\mu = 1$. Sin embargo sería mejor obtener más bins en la zona de $\mu = 1$ y menos donde hay menos estadística.

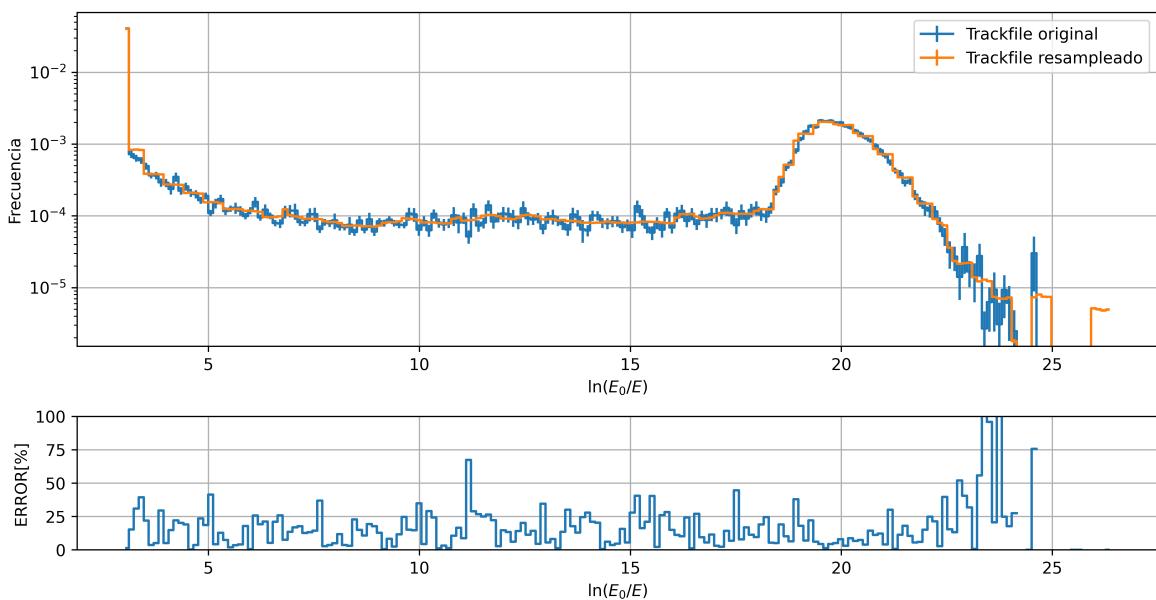


Figura 4.7: Comparación de la distribución de letargia entre el trackfile original y el trackfile remuestreado. Se observa el efecto de separar manualmente los neutrones con letargia mínima.

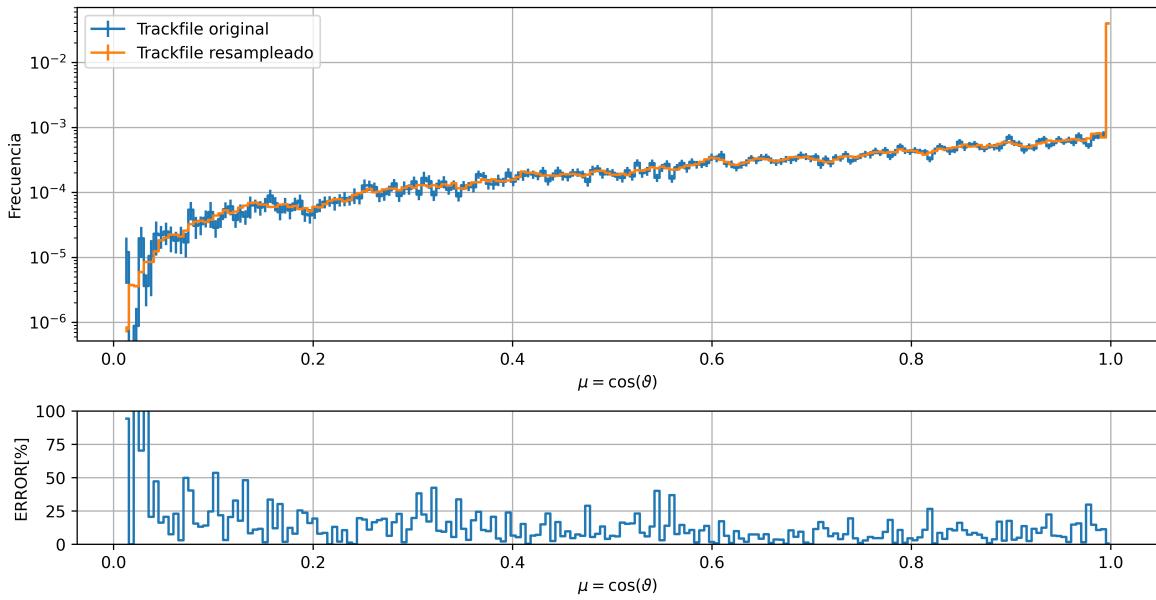


Figura 4.8: Comparación de la distribución de μ entre el trackfile original y el trackfile remuestreado. Se observa el efecto de separar manualmente los neutrones con $\mu = 1$.

A su vez en el grafico 2D de x vs y se observa que la distribucion del canal de vacio se mantiene mas concentrada y no se difumina en el agua. Sin embargo sigue habiendo una forma de cruz en el agua. Esto se sigue debiendo a la discretizacion uniforme de macrogupos. Para solucionar esto deberia haber mas macrogupos cerca de la interface y menos cerca del borde del modelo, acorde a la cantidad de estadistica presente en el *trackfile* original. En la figura 4.9 se observa la distribucion de x vs y para el *trackfile* remuestreado utilizando discretizacion uniforme tanto para macrogupos como para microgrupos, pero con bordes manuales.

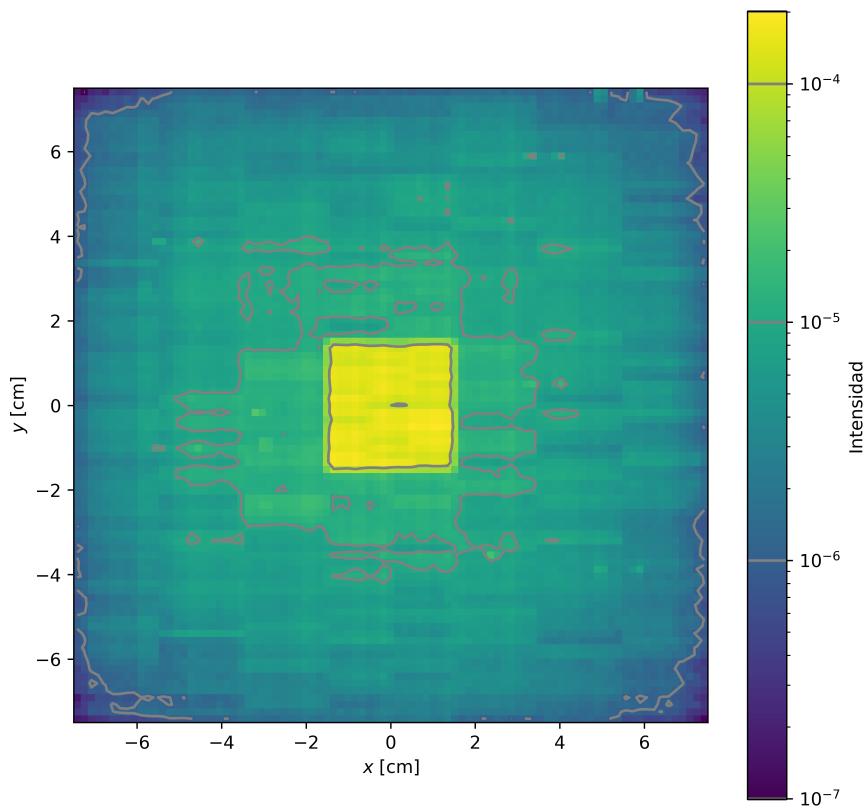


Figura 4.9: Distribuciones de x vs y para el *trackfile* remuestreado utilizando bordes manuales para separar el canal de vacío explicitamente.

En la tabla ?? se presentan las divergencias KL 1D y 2D, tanto de forma parcial como total, para la configuración de bineado uniforme con bordes manuales. Comparando estos valores contra los valores de la tabla 4.1 referidos a la configuración de bineado uniforme sin bordes manuales, se observa que la divergencia KL ha disminuido en todos los casos. Esto indica que la segmentación manual de los bordes de los macrogrupos ha permitido mejorar la representación de las distribuciones y correlaciones originales. Los valores más destacados son los que contienen a la letargia y μ , debido a que concentraban la mayor estadística.

Tabla 4.2: Divergencia KL parcial y total para la configuración Equal / Equal.

KL 1D		KL 2D	
Parámetro	Valor	Parámetros	Valor
$\ln(E_0/E)$	5.4229e-01	$\ln(E_0/E), x$	2.7811e+00
x	3.3431e-01	$\ln(E_0/E), y$	2.7587e+00
y	3.4260e-01	$\ln(E_0/E), \mu$	1.1715e+01
μ	1.3672e+01	$\ln(E_0/E), \phi$	2.4997e+00
ϕ	4.1083e-01	x, y	2.4264e+00
		x, μ	1.1357e+01
		x, ϕ	2.1108e+00
		y, μ	1.1365e+01
		y, ϕ	2.0767e+00
		μ, ϕ	1.1336e+01
Suma KL 1D		Suma KL 2D	
1.5302e+01		6.0426e+01	

$$\text{KL total} = 7.5728e+01$$

4.2.5. Bineado adaptativo de microgrupos y macrogrupos

Posteriormente se aplicó un esquema de histogramas adaptativos, donde la subdivisión de los macrogrupos fue determinada de forma automática en función de la densidad estadística. Esta técnica permitió una segmentación más eficiente, sin requerir intervención manual del usuario para indicar los cambios bruscos más representativos.

Para obtener una comparación representativa contra el caso de bineado uniforme, se utilizó la misma cantidad de macrogrupos y microgrupos que en la configuración anterior. En este caso, se emplearon $n_{\text{macro}} = 6$ y $n_{\text{micro}} = 50$ para cada variable, en el orden de variables: $[\ln(E_0/E), x, y, \mu, \phi]$.

Analizando la distribución remuestreada de letargia y μ podemos ver que se obtiene una densidad de bins mayor en las regiones donde hay mayor estadística. En la figura 4.10 se observa que la distribución remuestreada conserva la delta en letargia mínima y se le asignan más bins a la zona de letargia baja y termalización. A su vez se observa que se asignan relativamente pocos bins en donde hay pocos bins, provocando una mayor suavidad. En la figura 4.11 se observa que la distribución remuestreada conserva la delta en $\mu = 1$ y se le asignan más bins a la zona de $\mu = 1$. Sin embargo se observa que se reproduce el ruido estadístico debido a que hay un exceso de micro bins.

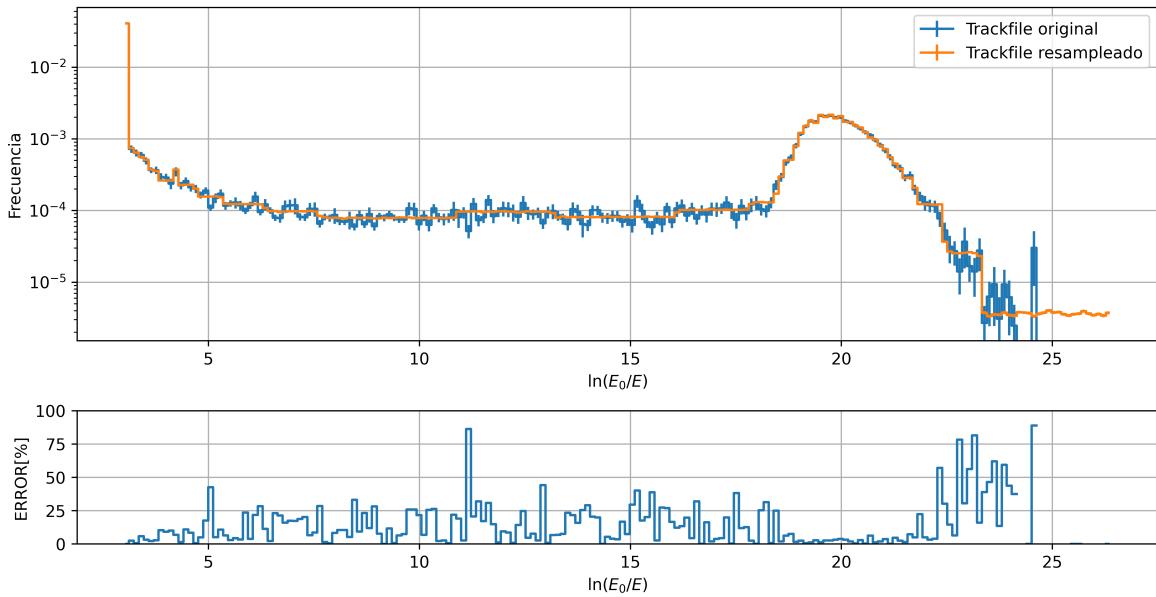


Figura 4.10: Comparación de la distribución de letargia entre el trackfile original y el trackfile remuestreado.

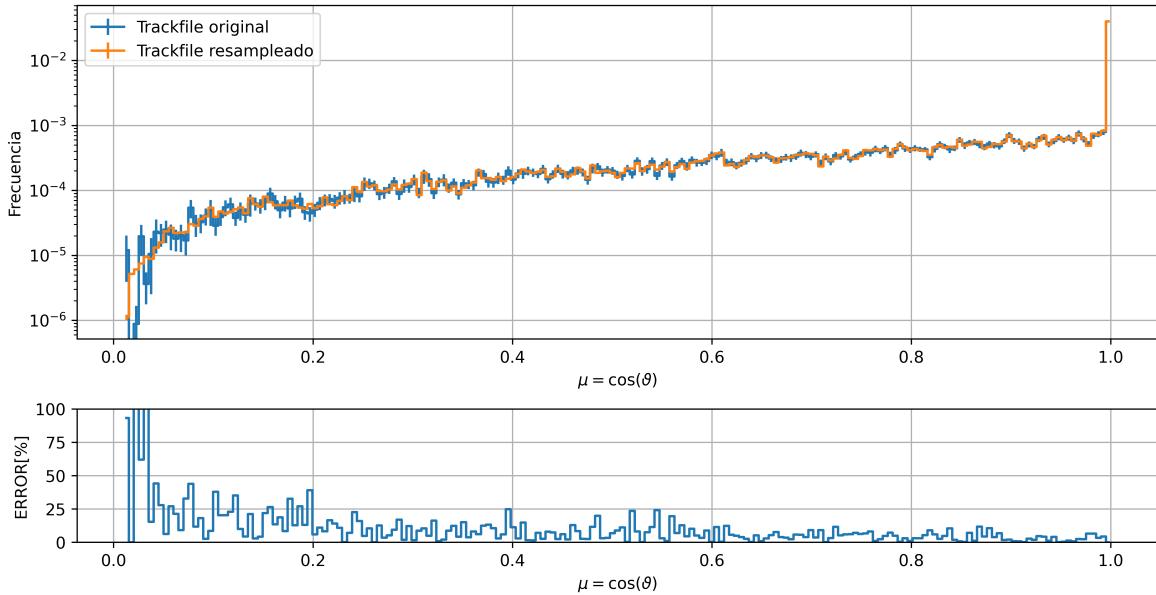


Figura 4.11: Comparación de la distribución de μ entre el trackfile original y el trackfile remuestreado.

Observando la distribución remuestreada de x vs y graficada en la figura 4.12 se observa que la distribución del canal de vacío se mantiene más concentrada y no se difumina en el agua. A su vez se observa que la distribución del agua se mantiene más difusa y no se observa la forma de cruz que se observaba en el caso de bineado uniforme. Esto es debido a que al ser un bineado adaptativo, los bins se asignan donde hay mayor estadística, evitando formar grupos de neutrones con poca estadística.

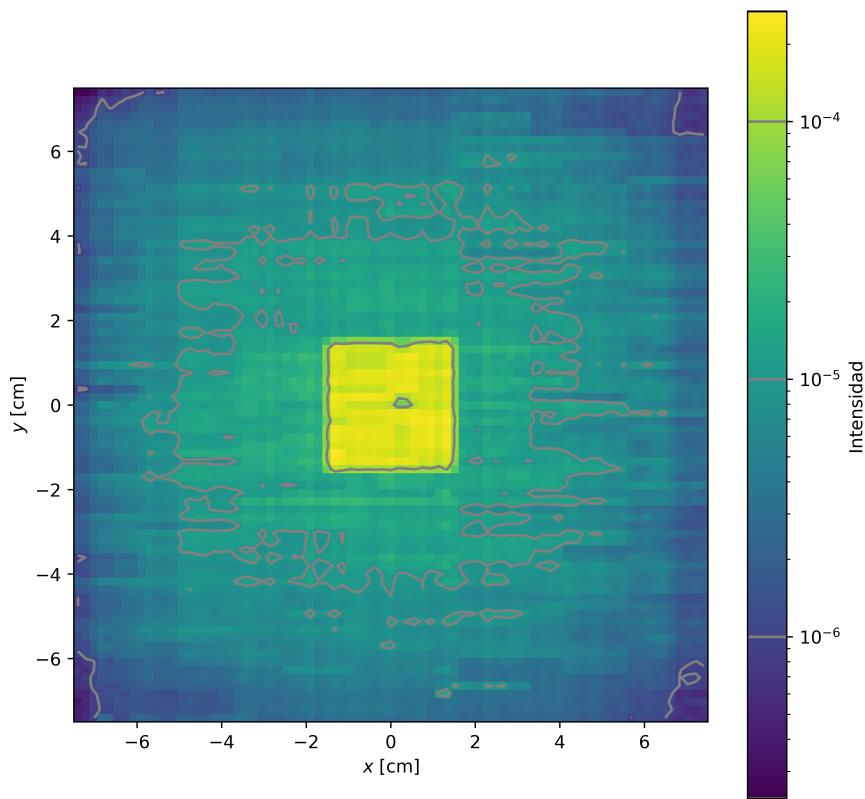


Figura 4.12: Distribuciones de x vs y para el *trackfile* remuestreado.

En la tabla 4.3 se presentan las divergencias KL 1D y 2D, tanto de forma parcial como total, para la configuración de bineado adaptativo. Comparando estos valores contra los valores de la tabla ?? referidos a la configuración de bineado uniforme con bordes manuales, se observa que se obtiene una métrica KL similar al caso de bineado uniforme con bordes manuales.

Tabla 4.3: Divergencia KL parcial y total para la configuración Equal / Equal.

KL 1D		KL 2D	
Parámetro	Valor	Parámetros	Valor
$\ln(E_0/E)$	5.4758e-01	$\ln(E_0/E), x$	2.8433e+00
x	3.4476e-01	$\ln(E_0/E), y$	2.9157e+00
y	3.5721e-01	$\ln(E_0/E), \mu$	1.1745e+01
μ	1.3679e+01	$\ln(E_0/E), \phi$	3.8279e+00
ϕ	4.3238e-01	x, y	2.7147e+00
		x, μ	1.1421e+01
		x, ϕ	3.1527e+00
		y, μ	1.1436e+01
		y, ϕ	3.0822e+00
		μ, ϕ	1.1630e+01
Suma KL 1D	1.5361e+01	Suma KL 2D	6.4769e+01

$$\text{KL total} = 8.0129e+01$$

4.2.6. Bineado adaptativo de microgrupos y macrogrupos optimizado

Debido a que la sección anterior mostro que se obtiene una mejor utilizacion de los bins al utilizar un bineado adaptativo, se decidió seguir trabajando con este método y optimizar la cantidad de microgrupos y macrogrupos. Es posible aumentar la cantidad de macrogrupos y disminuir la cantidad de microgrupos, obteniendo una mejor representación de las correlaciones y una distribución de las variables más suavizada en las regiones de menor estadística. A su vez es posible utilizar una cantidad de macrogrupos y microgrupos distinta para cada variable.

Se utilizó la siguiente configuración de macrogrupos y microgrupos:

- **Macrogrupos:** se utilizó la configuración [6, 6, 6, 6, 6].

- **Microgrupos:** se utilizó la configuración [50, 50, 50, 50, 50].

4.3. Resultados de la simulación comparativa

4.3.1. Validación de tallies y aplicación de técnicas de reducción de varianza

Para validar los resultados se evaluaron distintas magnitudes físicas a lo largo del eje del sistema:

- Flujo escalar en secciones transversales: total, en agua, y en vacío.
- Espectro energético sobre una superficie de tally a $z = 80$ cm.
- Corriente en dirección z sobre planos intermedios.

Se aplicó reducción de varianza mediante weight windows generados con OpenMC, especialmente en regiones con moderación intensa, lo cual mejoró la estadística de tallies en el agua sin alterar el resultado global.

4.3.2. Síntesis y conclusiones

El caso del canal de vacío embebido en agua permitió poner en evidencia los desafíos que presentan las técnicas de muestreo cuando coexisten poblaciones de partículas con comportamientos disímiles. Se comprobó que:

- La segmentación espacial y direccional es crucial para preservar correlaciones en el muestreo.
- Los histogramas adaptativos constituyen una alternativa robusta y automática frente a configuraciones manuales.
- El método propuesto reproduce con alta fidelidad los resultados de flujo, espectro y corriente.

Este caso sirve como referencia para futuras aplicaciones en geometrías más complejas donde también existan discontinuidades materiales o comportamientos multimodales del campo de neutrones.

Capítulo 5

Validación Experimental: Conducto Número 5 del Reactor RA-6

- 5.1. Descripción del caso experimental del RA-6
- 5.2. Optimización del resampleo para la fuente experimental
- 5.3. Resultados de simulación en OpenMC y comparación experimental
 - 5.3.1. Distribución espacial del flujo en detectores
 - 5.3.2. Comparación ${}^3\text{He}$: simulación vs. experimento
- 5.4. Discusión sobre resultados y validación

Apéndice A

Implementación Computacional del Método

A.1. Código detallado en Python/C++

A.2. Pseudocódigo y comentarios

A.3. Archivos de configuración XML

A.4. Flujo de trabajo práctico

Bibliografía

- [1] Fairhurst Agosta, R. Cálculo neutrónico detallado de haces y guías de neutrones del reactor RA-10. Proyecto integrador de ingeniería nuclear, Instituto Balseiro, Centro Atómico Bariloche, 2017. [5](#)
- [2] Ayala, J. E. Implementación de una línea de cálculo basada en el código TRIPOLI a problemas de blindaje del reactor RA-10. Proyecto integrador de ingeniería nuclear, Instituto Balseiro, Centro Atómico Bariloche, 2019.
- [3] Abbate, O. I. Cálculo de blindajes mediante fuentes de distribución en McStas y Tripoli. Proyecto integrador de ingeniería nuclear, Instituto Balseiro, Centro Atómico Bariloche, 2020. [5](#)
- [4] Abbate, O. I. KDSource: Desarrollo de una herramienta computacional para el cálculo de blindajes. Tesis de maestría, Instituto Balseiro, Centro Atómico Bariloche, 2021. [5](#)
- [5] Schmidt, N., Abbate, O. I., Prieto, Z. M., Robledo, J. I., Márquez, J. I., Márquez, A. A., *et al.* Kdsource, a tool for the generation of monte carlo particle sources using kernel density estimation. *Annals of Nuclear Energy*, **177**, 109309, 2022.
- [6] Fox, F. Optimización de algoritmos de estimación de densidades para el cálculo de haces de neutrones y fotones. Proyecto integrador de ingeniería nuclear, Instituto Balseiro, Centro Atómico Bariloche, 2022.
- [7] Giménez, M. A. Implementación de muestreo on-the-fly en el código KDSource. Proyecto integrador de ingeniería nuclear, Instituto Balseiro, Centro Atómico Bariloche, 2024. [5](#)
- [8] OpenMC: Monte carlo code for nuclear simulation. <https://openmc.org/>, 2025. Último acceso: mayo de 2025. [7](#)
- [9] Documentación oficial de KDSource. <https://kdsouce.readthedocs.io/en/latest/index.html>, 2024. Último acceso: mayo de 2025. [8](#)

-
- [10] Hdf5®: The hierarchical data format. <https://docs.hdfgroup.org/archive/support/HDF5/doc/index.html>, 2025. Último acceso: mayo de 2025. 8
 - [11] MCPL: Monte carlo particle lists. <https://mctools.github.io/mcpl/>, 2025. Último acceso: mayo de 2025. 8
 - [12] Extensible markup language (xml) 1.0 (fifth edition). <https://www.w3.org/TR/xml/>, 2008. Último acceso: mayo de 2025. 9

Publicaciones asociadas

1. Mi primer aviso en la revista **ABC**, 1996
2. Mi segunda publicación en la revista **ABC**, 1997

Agradecimientos

A todos los que se lo merecen, por merecerlo

