

DESARROLLO DE TABLAS DE SECCIONES EFICACES DEPENDIENTES DE MÚLTIPLES PARÁMETROS EN DRAGON V5 PARA CNA2

Vignolo, R.¹ Giuntoli, G.¹ Khatchikian, F.²

¹TECNA Estudios y Proyectos de Ingeniería S.A.
Encarnación Ezcurra 365, C1107CLA Buenos Aires, Argentina
rvignolo@tecna.com

²Nucleoeléctrica Argentina S.A
Francisco Narciso de Laprida 3100, B1603AAA Vicente López, Argentina
fkhatchikian@na-sa.com.ar

Resumen

Con el fin de determinar las discrepancias encontradas entre el coeficiente de reactividad por temperatura del combustible (*doppler*) medido en la Central Nuclear Atucha II y el calculado a partir de secciones eficaces obtenidas mediante el código de celda WIMS de la forma usual, surgió la necesidad de modelar la celda elemental de CNA2 a través del código DRAGON, dado que este no sólo permite tener en cuenta ciertos fenómenos previamente considerados despreciables, sino que también posee una serie de funciones o procedimientos, lenguaje de *macros* y programas externos que facilitan la obtención de tablas de múltiples parámetros. El concepto de tablas de múltiples parámetros surge debido a que para representar adecuadamente éstos nuevos fenómenos considerados, debe pasarse de una tabla usual consistente en un único parámetro de entrada (el quemado) a una de múltiples parámetros (por las nuevas dependencias no lineales). En este trabajo se presenta la teoría detrás de los parámetros globales y locales de secciones eficaces que ha permitido identificar cuáles de ellos deben ser seleccionados para representar adecuadamente la central nuclear, tanto en cálculos de núcleo estacionarios como en transitorios de cinética espacial. En este contexto, se describe la implementación de estos conceptos dentro de un *deck* de cálculo que utiliza DRAGON como motor de cálculo neutrónico y, posteriormente, se analizan los resultados obtenidos.

DEVELOPMENT OF MULTIPLE PARAMETERS CROSS SECTIONS TABLES IN DRAGON V5 FOR ATUCHA II NUCLEAR POWER PLANT

Vignolo, R.¹ Giuntoli, G.¹ Khatchikian, F.²

¹TECNA Estudios y Proyectos de Ingeniería S.A.
Encarnación Ezcurra 365, C1107CLA Buenos Aires, Argentina
rvignolo@tecna.com

²Nucleoeléctrica Argentina S.A
Francisco Narciso de Laprida 3100, B1603AAA Vicente López, Argentina
fkhatchikian@na-sa.com.ar

Abstract

In order to determine the discrepancies found between the fuel temperature coefficient (*doppler*) measured at the Atucha II Nuclear Power Plant (CNA2) and the one calculated based on WIMS cross sections in the usual way, the need to model the fuel assembly of CNA2 in DRAGON arose. This lattice code not only allows to consider certain phenomena previously neglected, but it also has a number of functions or procedures, *macros* language and external programs that facilitate the process of obtaining multiple parameters tables. Multiple parameters table concept arises because a single parameter table is not adequate for the representation of these new phenomena, since new non-linear dependences are needed. This paper addresses the theory behind global and local cross sections parameters that has been used to identify which parameters should be considered for steady-state and transient (spatial kinetics) calculations. In this context, the implementation of these concepts into a calculation deck that uses DRAGON as cell code is presented and, later on, certain results are analyzed.

Índice

1. Introducción	4
2. Parámetros globales y locales	4
3. Descripción de las entradas de DRAGON	5
3.1. Estructuras de datos	6
3.2. Procedimientos	7
3.3. Parámetros de entrada	7
3.4. Geometrías	7
3.5. <i>Tracking</i>	8
3.6. Quemado de referencia	9
3.7. Definición de biblioteca	10
3.8. Cálculo de transporte	10
3.9. <i>Linear loop</i>	11
3.10. <i>Nested loop</i>	12
4. Extracción de datos	13
5. Análisis de las dependencias globales y locales	14
6. Modelo detallado	18
6.1. Descripción	18
6.2. Generación	20

1. Introducción

Con el fin de determinar las discrepancias encontradas entre el coeficiente de reactividad por temperatura del combustible (*doppler*) medido en planta y el calculado a partir de secciones eficaces obtenidas mediante el código de celda WIMS de forma usual, surgió la necesidad de modelar la celda elemental de la Central Nuclear Atucha II a través de una herramienta que permita tener en cuenta ciertos fenómenos previamente considerados despreciables. Se optó por el código de celda DRAGON en su versión 5 (Ref. [1]).

A lo largo de este trabajo se pondrán en evidencia las diferencias con la forma clásica de computar las secciones eficaces y sus dependencias. Entre ellas se destacan el hecho de poder diferenciar perturbaciones globales de perturbaciones locales o tener en cuenta el tratamiento resonante de subgrupos en la pastilla combustible que permite apantallar las secciones eficaces microscópicas según un perfil de temperatura no uniforme. Para poder implementar adecuadamente estos fenómenos al esquema de secciones eficaces, debe pasarse de la tabla usual de un único parámetro de entrada (el quemado) a una de múltiples parámetros (dado que existen nuevas dependencias no lineales).

Sin realizar un profundo hincapié en los análisis realizados, se presentará una breve descripción del modelo de celda de CNA2 realizado en DRAGON. En otras palabras, si bien DRAGON posee diversas formas de representar la geometría de la celda combustible de CNA2, los motivos que llevaron a optar por la geometría de celda utilizada en este trabajo escapa del alcance del mismo. Esta geometría es el resultado de numerosos estudios donde se han optimizado diversos factores tanto para el problema que se quiere resolver como para el estado de arte actual del código.

En el contexto del desarrollo de estos nuevos modelos de secciones eficaces se optó por utilizar, dentro de lo posible, las últimas incorporaciones en el código de celda. Esto no sólo tiene que ver con la implementación de los últimos módulos de DRAGON, si no que además se utilizaron herramientas externas fuertemente recomendadas por los desarrolladores del código fuente. Dentro de ellas, la más importante ha sido el uso de las bibliotecas GanLib (Ref. [2]) que permiten extraer las secciones eficaces almacenadas dentro de las estructuras de datos generadas por el módulo COMPO. Como resultado se ha obtenido un *deck* de cálculo que, con gran armonía, combina la utilización de DRAGON, GanLib, wasora (Ref. [3]), xdfrrpf (Ref. [4]) y RELAP (Ref. [5]) a partir de ciertos *scripts*, permitiendo obtener tablas de secciones eficaces dependientes de múltiples parámetros.

2. Parámetros globales y locales

A la hora de definir un esquema para una tabla de secciones eficaces es necesario poder, entre otras cosas, reconocer las diferencias entre los parámetros de entrada globales y locales, identificar como repercute incorporar un nuevo parámetro en los tiempos de cálculo o en la cantidad de cálculos necesarios y establecer bajo que condiciones es posible aproximar una perturbación global con una local. Entonces, los parámetros se dividen en

- parámetros globales \mathbf{P}_g : son aquellos parámetros constantes durante el quemado; y
- parámetros locales \mathbf{P}_l : son aquellos parámetros que varían en un cierto quemado.

En las entradas de DRAGON se diferencian a los parámetros globales como parámetros de entrada y a los parámetros locales como parámetros de perturbación. De esta manera, los parámetros globales definen las composiciones de los materiales a cierto quemado Q_i , mientras que los parámetros locales determinan las propiedades y composiciones de los materiales a un cierto quemado.

Según lo discutido previamente, la sección eficaz macroscópica de cierto material es tanto función de su historia como de sus condiciones al momento de ser evaluada, $\Sigma_x(Q, \mathbf{P}_g, \mathbf{P}_1)$. Sin embargo, dado que no es práctico realizar un cálculo de celda para cada una de las posibles combinaciones de parámetros de entrada y de perturbación, es importante definir adecuadamente tanto los parámetros como algunas aproximaciones.

En primer lugar se supone que las perturbaciones son separables, es decir, una variación en temperatura y densidad puede ser reemplazada por una variación en temperatura a densidad constante y viceversa. En segundo lugar, cuando cierta dependencia de una sección eficaz con algún parámetro es lineal, puede reemplazarse la variación de la misma en un desarrollo de Taylor de primer orden. Ambas aproximaciones resultan en la siguiente aproximación para $\Sigma_x(Q, \mathbf{P}_g, \mathbf{P}_1)$:

$$\Sigma_x(Q, \mathbf{P}_g, \mathbf{P}_1) \approx \Sigma_x(Q, \mathbf{P}_g^0, \mathbf{P}_1^0) + \sum_{k=1}^{N_G} \left. \frac{\partial \Sigma_x}{\partial P_{g,k}} \right|_{(Q, \mathbf{P}_g^0, \mathbf{P}_1^0)} \cdot (P_{g,k} - P_{g,k}^0) + \sum_{k=1}^{N_L} \left. \frac{\partial \Sigma_x}{\partial P_{l,k}} \right|_{(Q, \mathbf{P}_g^0, \mathbf{P}_1^0)} \cdot (P_{l,k} - P_{l,k}^0). \quad (1)$$

Comúnmente, el término $\Sigma_x(Q, \mathbf{P}_g^0, \mathbf{P}_1^0)$ se conoce como valor central de tabla y hace referencia a la sección eficaz evaluada a cierto quemado y combinación de parámetros globales iguales a los locales (es decir, una condición sin perturbar). Sin embargo, el valor central de tabla puede tomar en cuenta la dependencia con alguno de los parámetros globales o locales con el que no se presenta una dependencia lineal. En el caso de tomar en cuenta la dependencia con algún parámetro local, el valor central de tabla no coincide con una condición sin perturbar. Por otra parte, aquellos parámetros con los que se presentan dependencias lineales son tenidos en cuenta en ambas sumatorias. La sección 3 explica la lógica utilizada en DRAGON para generar tanto los valores centrales de tabla como las derivadas de la ecuación 1, mientras que la sección 5 vuelca los resultados obtenidos con el fin de poder definir posibles esquemas de tablas de secciones eficaces.

3. Descripción de las entradas de DRAGON

En esta sección se describe la lógica detrás de los *inputs* realizados en DRAGON que generan los datos necesarios para fabricar las tablas de secciones eficaces dependientes de múltiples parámetros. El algoritmo 1 presenta un primer esquema que será desglosado en las próximas secciones al detallar, entre otras cosas, los procedimientos desarrollados. La comprensión de estos algoritmos es relevante debido a que posteriormente son incorporados dentro de un *deck* de cálculo que permite obtener las secciones eficaces para ciertos parámetros definidos. De esta forma, diferentes modelos pueden ser propuestos pero utilizando en todos ellos, a fin de cuentas, las estructuras elementales aquí descriptas.

Algoritmo 1: Descripción de un *input* característico en DRAGON.

Datos: Características de la celda combustible de CNA2.

Resultado: Archivo ASCII con jerarquía tipo GanLib conteniendo las secciones eficaces pertinentes.

declarar estructuras de datos (LINKED_LISTs);

declarar archivos SEQ_ASCII y SEQ_BYNARY;

declarar procedimientos;

procedimiento DefParams: definir parámetros de entrada;

definir vectores de quemado, apantallamiento y perturbaciones locales;

procedimiento DefGeos: declarar geometrías;

tracking o *ray tracing* de geometrías;

imprimir salidas gráficas: *mixtures & regions*;

inicializar multicompo *linked list*;

realizar quemado de referencia;

si el tipo de perturbación es *nested* entonces

 | realizar perturbaciones anidadas;

en otro caso

 | realizar perturbaciones lineales;

3.1. Estructuras de datos

Una estructura de datos de DRAGON es una estructura que contiene datos de cierto tipo, dependiendo qué módulo la originó. Permiten el pasaje de información entre diferentes módulos y, según se busque optimizar el tiempo de cálculo o los recursos de memoria, pueden estar como *linked lists* (*memory resident*) o como *xsm files* (*memory persistent*). Para poder comprender los pseudocódigos descriptos en las siguientes secciones es necesario reconocer las *linked lists* utilizadas:

- *miclib*: estructura de datos que contiene las secciones eficaces microscópicas y macroscópicas para transferir entre módulos;
- *miclib_s*: estructura de datos que contiene secciones eficaces microscópicas y macroscópicas luego del tratamiento resonante para transferir entre módulos;
- *geo_s*: estructura de datos que contiene la geometría para el cálculo de apantallamiento;
- *geo_t*: estructura de datos que contiene la geometría para el cálculo de transporte;
- *track_s*: estructura de datos que contiene los datos del *ray tracing* de la geometría para apantallamiento;
- *track_t*: estructura de datos que contiene los datos del *ray tracing* de la geometría para transporte;
- *burnup*: estructura de datos que almacena la información del depletado; y
- *flux*: estructura de datos para transferir el flujo calculado entre módulos.

3.2. Procedimientos

Se han realizado procedimientos dado que la utilización de ellos simplifica las entradas de DRAGON, disminuyendo considerablemente la probabilidad de cometer errores. Los procedimientos de DRAGON tienen como objetivo la misma utilidad que una función en un código: resolver determinado problema dado ciertos parámetros de entrada. En este sentido, se han realizado cuatro procedimientos que serán abordados en las próximas secciones:

- DefParams: procedimiento que permite definir diversos parámetros de entrada,
- DefGeos: procedimiento que permite definir las geometrías a emplear en el cálculo,
- DefLib: procedimiento que permite definir la biblioteca, y
- CalcFlux: procedimiento que realiza un cálculo de transporte.

3.3. Parámetros de entrada

Según lo discutido en la sección 2, los parámetros de entrada se corresponden con los parámetros globales en una corrida de DRAGON. Estos y otros parámetros se encuentran definidos en el procedimiento DefParams.c2m, descrito en el algoritmo 2.

Algoritmo 2: Definición de parámetros mediante procedimiento DefParams.

inicio

comienzo de la definición de parámetros;
 definir parámetros para la biblioteca (parámetros globales);
 definir parámetros para las geometrías;
 definir parámetros para el *tracking* o *ray tracing*;
 definir parámetros para el quemado;
 retornar parámetros;

3.4. Geometrías

La celda combustible elemental de CNA2 puede ser descripta de diversas formas utilizando diferentes instrucciones de DRAGON (ver Ref. [6]). Sin embargo, aquí se presentarán únicamente dos geometrías, una detallada a partir de una celda hexagonal triangular con *clusters* y una menos detallada a partir de una celda hexagonal con *clusters*. El algoritmo 3 describe la forma la que el procedimiento DefGeos.c2m define las geometrías mientras que las figuras 1 y 2 presentan los resultados.

La geometría más simple es capaz de reproducir resultados semejantes a aquellos obtenidos con la geometría detallada, pero reduciendo considerablemente el tiempo de cálculo. Esto se debe a que se han optimizado, entre otras cosas, la discretización espacial y el mallado azimutal de los *pines*.

Algoritmo 3: Definición de geometrías mediante procedimiento DefGeos.

inicio

comienzo de la definición de geometrías;
 leer parámetros de entrada para las geometrías;
 definir geometría para apantallamiento de σ ;
 definir geometría para la resolución de transporte;
 retornar *geo_s* y *geo_t* *linked lists*;
 fin

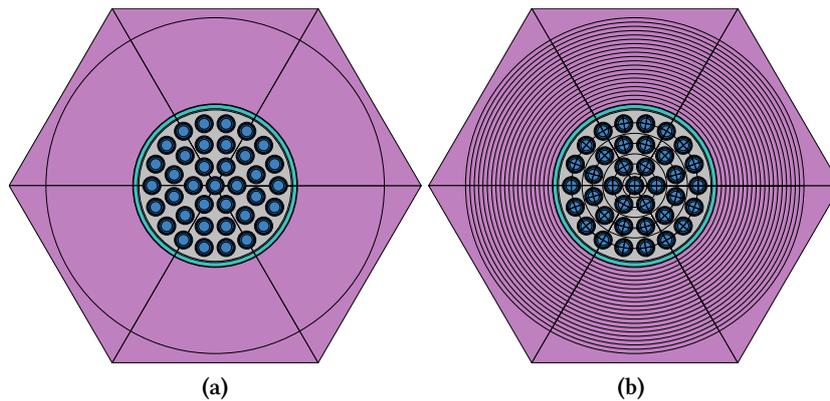


Figura 1: Geometrías HEXCEL para el cálculo de apantallamiento *geo_s* (a) y de transporte *geo_t* (b).

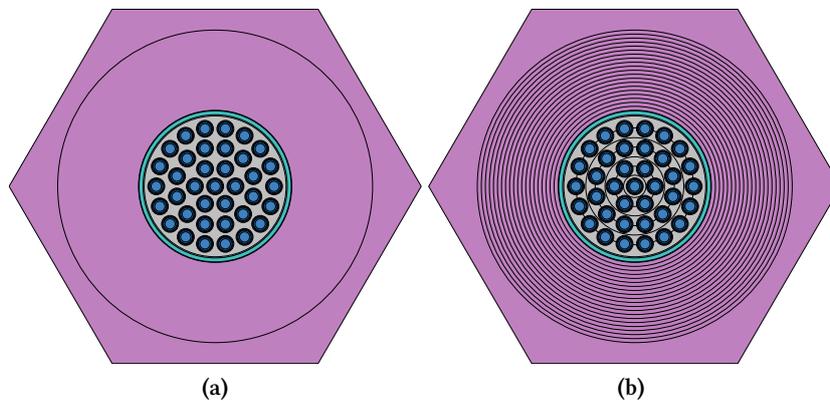


Figura 2: Geometrías HEXCEL para el cálculo de apantallamiento *geo_s* (a) y de transporte *geo_t* (b).

3.5. Tracking

Como ya se ha mencionado, son dos las geometrías que necesitan ser procesadas por un módulo de *ray tracing* de DRAGON: una a utilizar en los cálculos de apantallamiento y otra en los de

transporte. Si bien el *ray tracing* debe realizarse una única vez, dependiendo de la cantidad de información generada por este, otros módulos verán su tiempo de cálculo afectado considerablemente. Por ejemplo, la cantidad de probabilidades de colisión depende del número de regiones definidas o la cantidad de términos que contribuyen al cálculo de cada probabilidad de colisión depende de la densidad de rayos y de la cantidad de ángulos (ver Ref. [7]). Por este motivo, es de suma importancia optimizar tanto las regiones como las densidades y los ángulos de las líneas del *ray tracing*. La optimización de la densidad y los ángulos de las líneas resultó en 5 cm^{-1} y 5° respectivamente, tal como puede verse en la figura 3.

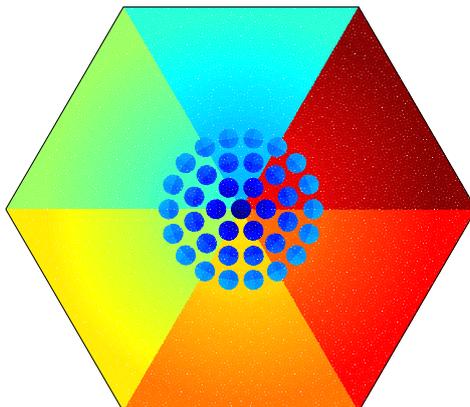


Figura 3: Líneas del *ray tracing* empleado en las geometrías de apantallamiento y transporte.

3.6. Quemado de referencia

El quemado de referencia es un artilugio que permite realizar, en primera instancia, una evolución en quemado con ciertos parámetros globales \mathbf{P}_g fijos. La información del depletado es almacenada en la estructura de datos burnup con el fin de poder realizar su posterior uso en un *linear* (sec. 3.9) o *nested loop* (3.10). La lógica detrás de un quemado de referencia se presenta en el algoritmo 4.

Algoritmo 4: Quemado de referencia.

inicio

```

comienzo del quemado de referencia: reference burnup loop;
procedimiento DefLib: definir biblioteca;
procedimiento CalcFlux: apantallar y calcular transporte inicial (crear flux linked list);
imprimir salida gráfica: fluxes;
inicializar burnup linked list;
condensar en energía y homogeneizar espacialmente;
para cada paso de quemado
    guardar información en burnup linked list (tiempo inicial);
    depletar y posteriormente actualizar biblioteca al nuevo paso de quemado;
    procedimiento CalcFlux: apantallar y calcular transporte (modificar flux linked list);
    condensar en energía y homogeneizar espacialmente;
    guardar información en burnup linked list (tiempo final);

```

3.7. Definición de biblioteca

El procedimiento DefLib.c2m tiene como función definir tanto los materiales como sus propiedades. Se encuentra preparado para soportar bibliotecas con formato DRAGLIB (por ejemplo, SHEM) o WIMSD4 (por ejemplo, WLUP).

Algoritmo 5: Definición de biblioteca mediante procedimiento DefLib.

inicio

```

comienzo de la definición de la biblioteca;
leer parámetros de entrada para la biblioteca;
si la biblioteca es WIMSD4 entonces
  | definir materiales con nomenclatura WIMSD4 (saturados o a evolucionar según
  | corresponda);
sinó, si la biblioteca es DRAGLIB entonces
  | definir materiales con nomenclatura DRAGLIB (saturados o a evolucionar según
  | corresponda);
en otro caso
  | abortar;
retornar miclib linked list;

```

3.8. Cálculo de transporte

El procedimiento CalcFlux.c2m permite resolver la ecuación de transporte mediante el método de las probabilidades de colisión obteniendo como resultado la distribución espacial y en energía del flujo escalar y el factor de multiplicación. El pseudocódigo detrás de ésta instrucción se detalla en el algoritmo 6, mientras que la figura 4 representa la distribución espacial del flujo para un grupo rápido y otro térmico.

Algoritmo 6: Cálculo de transporte mediante procedimiento CalcFlux.

inicio

```

comienzo del cálculo de transporte;
leer parámetros de entrada para el cálculo;
si corresponde apantallar entonces
  | si es el primer paso de quemado entonces
  | | apantallar y crear miclib_s linked list;
  | en otro caso
  | | apantallar y modificar miclib_s linked list;
calcular probabilidades de colisión;
si es el primer paso de quemado entonces
  | calcular transporte y crear flux linked list;
en otro caso
  | calcular transporte y modificar flux linked list;
extraer y retornar  $k_{\infty}$ ;
retornar miclib_s y flux linked lists;

```

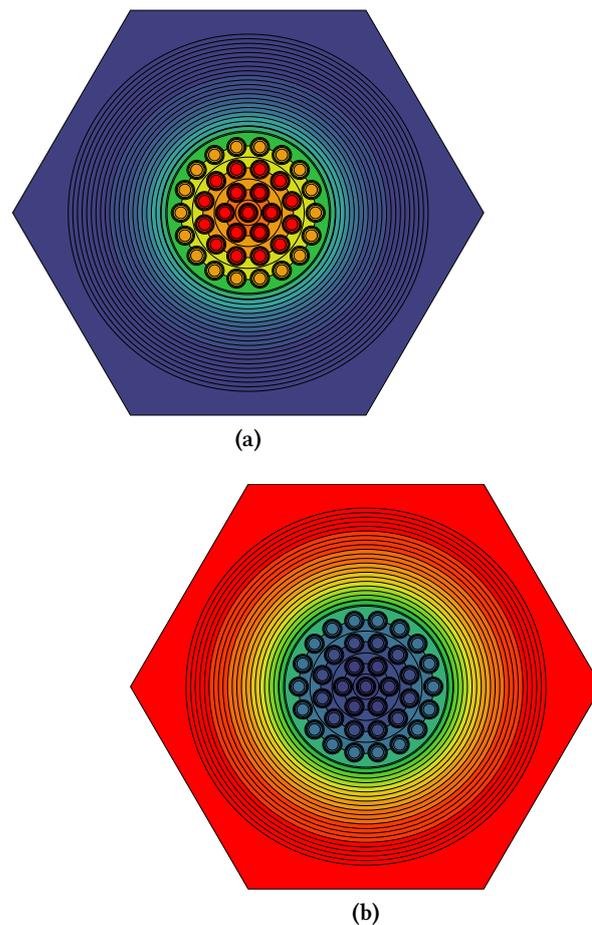


Figura 4: Distribución espacial del flujo pertenecientes al grupo más rápido (a) y más térmico (b) sin quemar.

3.9. *Linear loop*

Un *linear loop* permite realizar las perturbaciones locales sobre algún parámetro de interés con el fin de obtener, para cada paso de quemado Q_i , la dependencia de las secciones eficaces con el parámetro en cuestión. Los algoritmos 7 y 8 presentan la lógica detrás de una perturbación de este tipo para dos casos de interés: perturbación sobre un parámetro termohidráulico o sobre la concentración de xenón.

Como se detalla en los algoritmos 7 y 8, los resultados de un *linear loop* se almacenan en un objeto multicompo. De esta manera, con un *linear loop* se obtiene, para un mismo vector de parámetros globales \mathbf{P}_g^0 , la dependencia de cada sección eficaz con el quemado y el parámetro local, es decir, $\Sigma_x(Q, P_l)$. Si bien la sección eficaz no varía linealmente con el quemado, puede ocurrir que si lo haga con el parámetro local. En este caso, para disminuir la dimensión de la interpolación del valor central de tabla $\Sigma_x(Q, \mathbf{P}_g^0, \mathbf{P}_l^0)$, se calcula y tabula la derivada $\partial\Sigma_x/\partial P_l$ en función del quemado.

Algoritmo 7: Linear loop para perturbación de parámetro termohidráulico.

inicio

```

comienzo de las perturbaciones lineales: linear loop;
para cada perturbación local de parámetro termohidráulico
  procedimiento DefLib: definir biblioteca con parámetro perturbado;
  para cada paso de quemado
    si es el primer paso de quemado entonces
      procedimiento CalcFlux: apantallar y calcular transporte (crear flux linked list);
    en otro caso
      actualizar concentraciones al paso de quemado correspondiente;
      procedimiento CalcFlux: apantallar y calcular transporte (modificar flux linked list);
    condensar en energía y homogeneizar espacialmente;
    alimentar multicompo linked list;
  eliminar miclib, miclib_s y flux linked lists;

```

Algoritmo 8: Linear loop para perturbación de xenón.

inicio

```

comienzo de las perturbaciones lineales: linear loop;
para cada perturbación local de la concentración de xenón
  procedimiento DefLib: definir biblioteca con parámetro perturbado;
  para cada paso de quemado
    si es el primer paso de quemado entonces
      procedimiento CalcFlux: apantallar y calcular transporte (crear flux linked list);
    en otro caso
      actualizar concentraciones al paso de quemado correspondiente;
      sobrescribir concentración de xenón al valor perturbado;
      procedimiento CalcFlux: apantallar y calcular transporte (modificar flux linked list);
    condensar en energía y homogeneizar espacialmente;
    alimentar multicompo linked list;
  eliminar miclib, miclib_s y flux linked lists;

```

3.10. *Nested loop*

Cuando la dependencia de la sección eficaz con ciertos parámetros locales no es ni separable ni lineal, es posible implementar un *nested loop*. Esto permite, para un mismo vector de parámetros globales \mathbf{P}_g^0 , definir exactamente a la sección eficaz, es decir, sin la necesidad aplicar alguna de las aproximaciones previamente descriptas en la ecuación 1. Por otra parte, con este tipo de *loop* se

obtienen las secciones eficaces para todas las combinaciones posibles dentro del rango de variación definido para cada parámetro, motivo por el cual es necesario utilizarlo únicamente en el caso que esto se requiera o implementar restricciones en el dominio (evitar el cálculo para ciertas combinaciones de parámetros que no posean sentido físico).

Algoritmo 9: *Nested loop* para n perturbaciones de parámetros termohidráulicos

```

inicio
  comienzo de las perturbaciones anidadas: nested loop;
  para cada perturbación local de parámetro termohidráulico  $p_1$ 
    para cada perturbación local de parámetro termohidráulico  $p_2$ 
      :
      para cada perturbación local de parámetro termohidráulico  $p_n$ 
        procedimiento DefLib: definir biblioteca con parámetros perturbados;
        para cada paso de quemado
          si es el primer paso de quemado entonces
            procedimiento CalcFlux: apantallar y calcular transporte (crear flux
              linked list);
          en otro caso
            actualizar concentraciones al paso de quemado correspondiente;
            procedimiento CalcFlux: apantallar y calcular transporte (modificar
              flux linked list);
          condensar en energía y homogeneizar espacialmente;
          alimentar multicompo linked list;
        eliminar miclib, miclib_s y flux linked lists;
  
```

4. Extracción de datos

Los datos generados a partir de quemados de referencia, *linear loops* o *nested loops* son almacenados en estructuras de datos denominadas, usualmente, objetos multicompo. Estos objetos poseen un formato de datos que respeta cierta estructura que puede ser leída y procesada por las bibliotecas GanLib.

El Programa Cinético Espacial (PCE, Ref. [8]) requiere de los siguientes datos para poder determinar tanto un estado estacionario de planta como la evolución de determinado transitorio:

- secciones eficaces condensadas y homogeneizadas, que incluyen coeficientes de difusión, absorciones, *scattering* y fisiones rápidas y térmicas. En total son doce los valores a detallar: D_1 , D_2 , Σ_{a1} , Σ_{12} , Σ_{21} , Σ_{a2} , $\nu\Sigma_{f1}$, $\nu\Sigma_{f2}$, $e\Sigma_{f1}$, $e\Sigma_{f2}$, Σ_{f1} y Σ_{f2} ;
- constantes del xenón y yodo para la determinación de la dinámica de los mismos. Estas incluyen seis valores: *yields* de fisión para el xenón y yodo (térmicos y rápidos) y las secciones eficaces microscópicas de absorción del xenón;

- derivadas de las secciones eficaces con respecto a los parámetros termohidráulicos y el xenón, que se determinan a partir del análisis de objetos `multicompo` con información acerca de las secciones eficaces detalladas en el primer ítem.

El algoritmo 10 detalla el pseudocódigo de las rutinas escritas en C para extraer la información almacenada en las estructuras de datos `multicompo` y generar archivos para definir funciones en formato `wasora`. Esto último es debido a que tanto la interpolación multidimensional de PCE o el análisis para calcular la dependencia lineal de las secciones eficaces con algún parámetro se realiza bajo esquemas que utilizan funciones de este tipo.

Algoritmo 10: Extracción de datos de objetos `multicompo`.

Datos: Archivo ASCII perteneciente a un objeto `multicompo` conteniendo las secciones eficaces pertinentes.

Resultado: Archivo ASCII capaz de ser utilizado para definir funciones de secciones eficaces en `wasora`.

abrir archivo ASCII conteniendo el objeto `multicompo`;

definir archivo ASCII de salida;

verificar la consistencia con el número de cálculos esperados en el objeto `multicompo`;

leer puntos de definición de los cálculos;

para cada calculo contenido en el objeto `multicompo`

 | imprimir puntos de definición;

 | **si el objeto `multicompo` contiene información de la dinámica del yodo y xenón entonces**

 | | imprimir las seis constantes para la dinámica del yodo y xenón;

 | **sinó, si el objeto `multicompo` contiene información de secciones eficaces perturbadas**

 | **entonces**

 | | imprimir las doce secciones eficaces;

5. Análisis de las dependencias globales y locales

Con el objetivo de desarrollar en mayor profundidad la ecuación 1 se presentan, a continuación, algunos conceptos que serán tenidos en cuenta a la hora de describir el esquema de la tabla de secciones eficaces detallada en la sección 6. Estos conceptos abarcan tanto la descripción de las diferentes aproximaciones así como también la repercusión de cada una de ellas en los cálculos a realizar en un código de celda.

En el caso más simple la dependencia de las secciones eficaces es solo función de un único parámetro P . La sección eficaz exacta para un dado estado corresponde a $\Sigma_x(Q, P_g, P_l)$. Obtener este valor en un código de celda requiere de un cálculo en el que se quemó con P_g y una vez en el paso de quemado correspondiente, se perturbe con P_l (notar que si $P_l = P_g$, se corresponde a un caso sin perturbar). Si la dependencia con el parámetro local es lineal, podemos reducir la dimensión de la interpolación del valor central de tabla (se elimina la dependencia con P_l) de forma que

$$\Sigma_x(Q, P_g, P_l) \approx \Sigma_x(Q, P_g) \Big|_{P_l^0} + \frac{\partial \Sigma_x}{\partial P_l} \Big|_{(Q, P_g, P_l^0)} \cdot (P_l - P_l^0). \quad (2)$$

Teniendo en cuenta los temas abordados previamente, la estimación de la derivada $\partial\Sigma_x/\partial P_l(Q, P_g, P_l^0)$ se realiza a partir de un *linear loop* sobre el parámetro en cuestión. A su vez, el nuevo valor central de tabla puede desdoblarse en dos términos, de manera que

$$\Sigma_x(Q, P_g) \Big|_{P_l^0} \approx \Sigma_x(Q) \Big|_{(P_g^0, P_l^0)} + \frac{\partial\Sigma_x}{\partial P_g} \Big|_{(Q, P_g^0, P_l^0)} \cdot (P_g - P_g^0). \quad (3)$$

En este caso, una forma de estimar la derivada $\partial\Sigma_x/\partial P_g(Q, P_g^0, P_l^0)$ es a partir de numerosos cálculos con diferentes parámetros globales, quemados de referencia y *linear loops*. Estos resultados permiten estimar tanto la derivada parcial en función del quemado con dirección al parámetro global como aquella con dirección al parámetro local. Las figuras 5–7 resumen gráficamente estos conceptos descriptos. La figura 5 presenta una sección eficaz a quemado nulo. En este caso se evidencia que, debido a que no hay evolución en quemado, el valor de la sección eficaz no depende del parámetro global (es decir, no hubo evolución de los materiales) y solo depende de la condición local. La figura 6 demuestra que la dependencia con el parámetro global cobra importancia a mayores quemados debido a que la determinación de los materiales difiere según el parámetro global utilizado para evolucionar. Por último, la figura 7 tabula, a modo de resumen, las dependencias de la derivada en sentido del parámetro local y de la derivada direccional ($\partial\Sigma_x/\partial P_l + \partial\Sigma_x/\partial P_g$) como funciones del quemado, pudiéndose ver fácilmente que a bajos quemados la derivada en el sentido del parámetro global es despreciable. De todas formas, a medida que el quemado incrementa, no es correcto desestimar la derivada en dirección del parámetro global. Este tema volverá a ser abordado más adelante en esta misma sección.

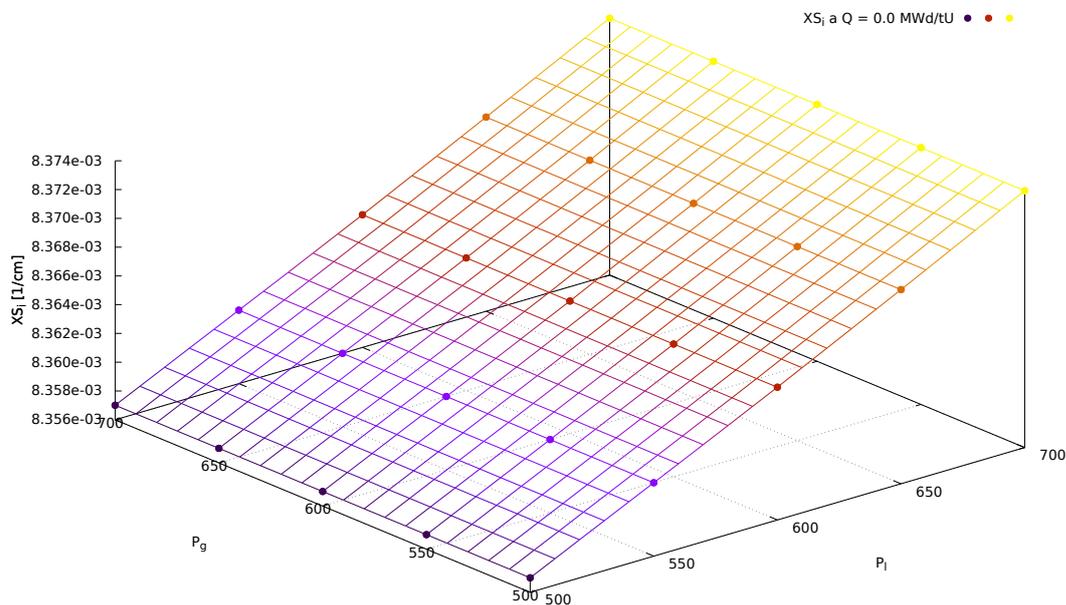


Figura 5: Esta figura presenta como es que la sección eficaz no varía en la dirección del parámetro global cuando se encuentra a $Q = 0$ MWd/tU.

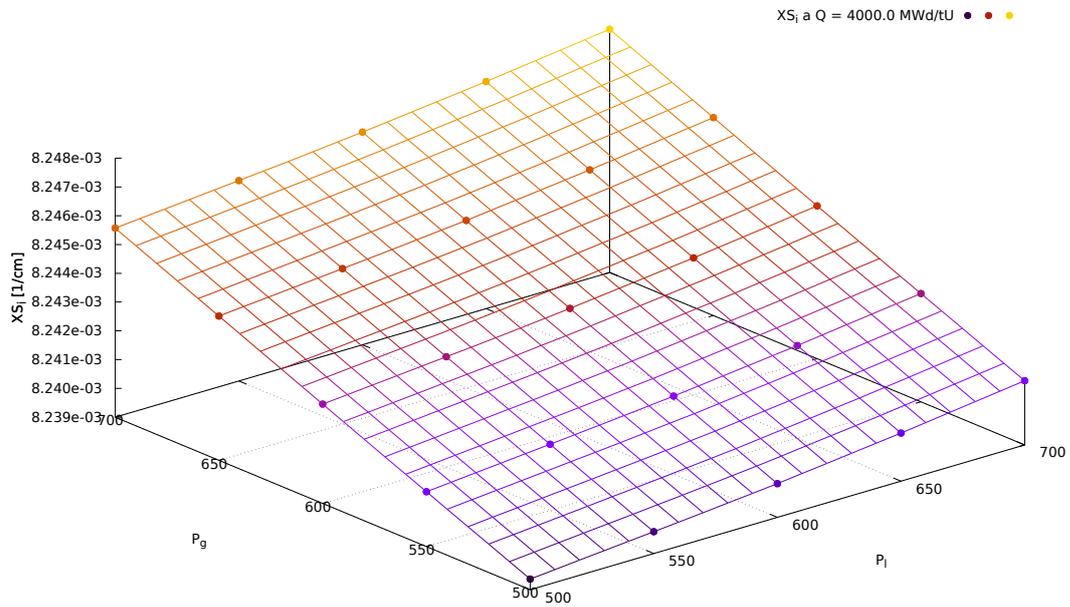


Figura 6: En este caso, a quemado $Q = 4000$ MWd/tU, la sección eficaz depende tanto del parámetro global como del local.

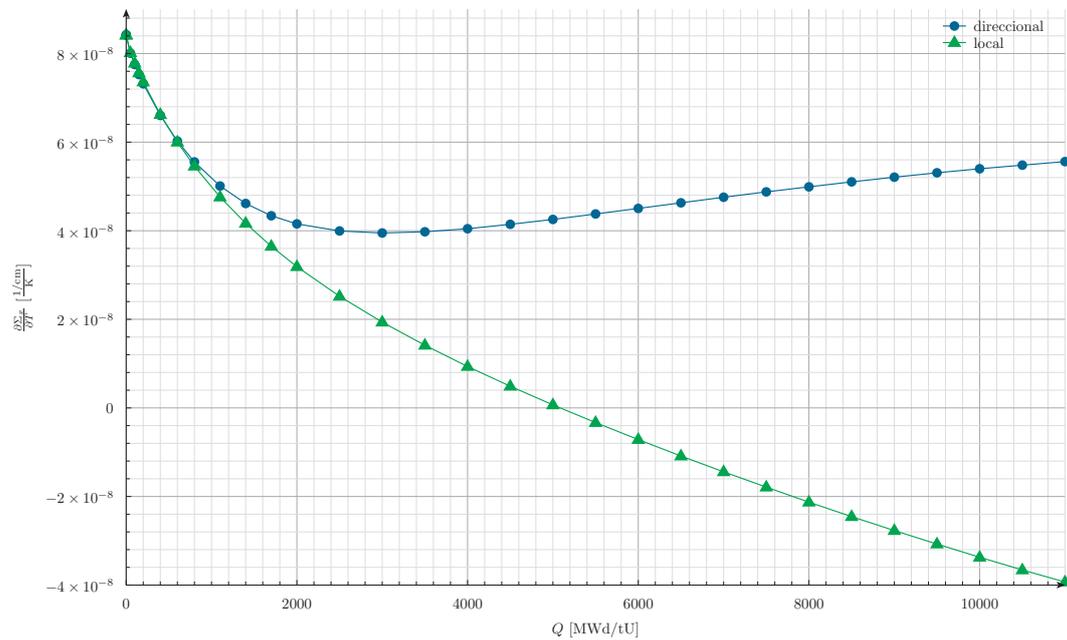


Figura 7: Derivada en la dirección del parámetro local y en la dirección de $P_g = P_l$ (direccional) de la sección eficaz en función del quemado. El hecho de que a bajos quemados ambas derivadas sean iguales refleja que la derivada en la dirección del parámetro global es despreciable. Posteriormente, esto deja de ser cierto, lo que indica que la derivada en el sentido del parámetro global no puede desestimarse.

Como ya se ha mencionado, en el caso que exista un parámetro con el que las secciones eficaces no varían linealmente, no es posible descomponer en un desarrollo de Taylor de primer orden al valor central de tabla. En este contexto, el valor central debe ser obtenido a partir de una interpolación que tenga en cuenta este parámetro.

En el caso general, la tabla de secciones eficaces posee una dependencia con el quemado y los vectores \mathbf{P}_g y \mathbf{P}_l . Traducir esto a un código de celda implica que, por cada combinación de parámetros globales, debe ejecutarse una corrida donde se realice un *nested loop* sobre los parámetros locales y sus rangos de variación. Con el fin de disminuir la cantidad de cálculos de un *nested loop* de gran dimensión, se aplica la aproximación desarrollada en la ecuación 2 en aquellos parámetros que pueden ser considerados de forma aislada y cuya dependencia puede incorporarse con derivadas estimadas a partir de *linear loops*. Sin embargo, aquellos parámetros con los que las secciones eficaces no presentan una dependencia lineal ni son separables deben incorporarse dentro del vector de interpolación, por lo que su dependencia es tabulada dentro del valor central. Por otro lado, con el fin de disminuir tanto la dimensión del vector de interpolación de la tabla como la cantidad de cálculos a realizar, la dependencia con los parámetros globales puede ser linealizada según la ecuación 3.

Es conveniente analizar dos situaciones de interés: la evaluación de las secciones eficaces durante la determinación de un estado estacionario de planta y durante un transitorio. En el caso estacionario se itera hasta encontrar las distribuciones de parámetros globales y locales, teniendo en cuenta que ambas son iguales, es decir, $\mathbf{P}_g(\mathbf{x}) = \mathbf{P}_l(\mathbf{x}) = \mathbf{P}(\mathbf{x})$. Si además se suponen perturbaciones separables y linealizables, la ecuación 1 resulta en

$$\Sigma_x^{\text{est}}(Q, \mathbf{P}_g^{\text{est}}, \mathbf{P}_l^{\text{est}}) \approx \Sigma_x(Q, \mathbf{P}^0, \mathbf{P}^0) + \sum_{k=1}^{N_P} \left(\frac{\partial \Sigma_x}{\partial P_{g,k}} + \frac{\partial \Sigma_x}{\partial P_{l,k}} \right) \Big|_{(Q, \mathbf{P}^0, \mathbf{P}^0)} \cdot (P_k^{\text{est}} - P_k^0), \quad (4)$$

donde la suma de ambas derivadas parciales da como resultado la derivada direccional en el sentido de $P_{g,k} = P_{l,k}$.

Durante un transitorio, la sección eficaz se evalúa a partir del valor obtenido en el estacionario más las perturbaciones locales que correspondan (suponiendo que éstas son separables y linealizables), es decir:

$$\Sigma_x^{\text{tr}}(Q, \mathbf{P}_g^{\text{est}}, \mathbf{P}_l^{\text{tr}}) \approx \Sigma_x^{\text{est}}(Q, \mathbf{P}_g^{\text{est}}, \mathbf{P}_l^{\text{est}}) + \sum_{k=1}^{N_P} \frac{\partial \Sigma_x}{\partial P_{l,k}} \Big|_{(Q, \mathbf{P}^0, \mathbf{P}^0)} \cdot (P_{l,k}^{\text{tr}} - P_{l,k}^{\text{est}}). \quad (5)$$

Analizando las ecuaciones 4 y 5, si la derivada en la dirección del parámetro global es despreciable frente a la derivada en la dirección del parámetro local, la determinación de la sección eficaz tanto en un estacionario y como en un transitorio se realiza de la misma forma:

$$\Sigma_x(Q, \mathbf{P}_g, \mathbf{P}_l) \approx \Sigma_x(Q, \mathbf{P}^0, \mathbf{P}^0) + \sum_{k=1}^{N_P} \frac{\partial \Sigma_x}{\partial P_{l,k}} \Big|_{(Q, \mathbf{P}^0, \mathbf{P}^0)} \cdot (P_{l,k} - P_{l,k}^0). \quad (6)$$

Sin embargo, esta última aproximación no ocurre para casi ninguno de los parámetros.

6. Modelo detallado

En esta sección se describen, en primera instancia, las consideraciones físicas y matemáticas del esquema detallado de secciones eficaces de CNA2. Por otro lado, y sólo a título informativo, se detalla la forma de obtener la tabla de múltiples parámetros a utilizar en PCE haciendo uso de diferentes programas y *scripts*. A su vez, con el fin de implementar los esquemas de generación de tablas de múltiples parámetros se incluye el desarrollo del pseudocódigo de cada uno de los *scripts*.

6.1. Descripción

La aplicación de la ecuación 1 a un modelo de reactor requiere seleccionar adecuadamente los parámetros y sus rangos de variación. Por este motivo, es de suma importancia estudiar la dependencia de las secciones eficaces con cada uno de los parámetros de entrada. El caso con mayor grado de detalle consiste en una tabla de secciones eficaces con los siguientes parámetros de entrada: quemado, temperaturas y densidades globales y locales del refrigerante y moderador, perfil de temperatura global y local del combustible y concentración de xenón global y local. La cantidad de combinaciones posibles entre estos parámetros equivale al producto de la discretización de cada uno de ellos. Obtener esta información no sólo requiere de una gran cantidad de cálculos de celda, si no que también de una costosa implementación computacional en un código de núcleo. Con el fin de calcular una tabla de secciones eficaces con este grado de detalle, pero evitando la costosa implementación, se seleccionaron los parámetros de entrada presentes en la tabla 1 y detallados a continuación.

	PL				PG		Funciones		
Q	θ_1	θ_2	θ_3	θ_4	q'	z	Σ_x	C_{Xe} y C_I	$(\partial\Sigma_x/\partial P)_{l,k}$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots

Cuadro 1: Esquema conceptual de la tabla de secciones eficaces para el modelo detallado.

- PL: parámetros locales;
- PG: parámetros globales;
- Q : quemado;
- θ_i : temperatura media local y adimensional del volumen anular i del combustible, tal que

$$\theta_i = \frac{T_i - 273.15 \text{ K}}{T_i^{\text{est}} - 273.15 \text{ K}};$$

- q' : potencia lineal global;
- z : representa la posición axial en el núcleo con el fin de determinar temperaturas y densidades globales y estacionarias del refrigerante y moderador¹. Por este motivo, previamente deben conocerse las distribuciones estacionarias de temperaturas y densidades, lo que implica que el proceso para generar esta tabla es iterativo entre la determinación de un estado estacionario de núcleo y los cálculos de celda;

¹El parámetro z toma como valor un entero entre 1 y 10 que permite determinar las propiedades medias correspondientes a los trozos $tr_1 = (2z - 1)$ y $tr_2 = 2z$ del reticulado de representación del reactor modelado en PCE.

- Σ_x : valor central de tabla;
- C_{Xe} y C_I : constantes para determinar la dinámica del xenón y yodo; y
- $(\partial\Sigma_x/\partial P)_{l,k}$: derivadas locales de las secciones eficaces respecto al parámetro P_k : xenón, temperatura o densidad del refrigerante y moderador.

En primer lugar se analizan los parámetros globales dejando a un lado los locales. Los parámetros globales son la potencia lineal q' y una magnitud z que representa la posición axial en el núcleo. Esta posición axial permite determinar el valor de las temperaturas y densidades globales del refrigerante y moderador si se conoce la distribución axial estacionaria de los mismos. A su vez, la temperatura del refrigerante junto a la potencia lineal y un modelo de combustible son utilizados para estimar el perfil de temperatura dentro de la pastilla (Ref. [9]). De esta forma, la elección de estos dos parámetros globales permite estimar a todos los parámetros globales reduciendo considerablemente los tiempos de cálculo debido a que cada combinación posible de parámetros globales \mathbf{P}_g requiere de una ejecución del código de celda con el correspondiente quemado de referencia y, según corresponda, un *nested* o *linear loop* para las perturbaciones locales.

La determinación del estado estacionario se lleva a cabo únicamente a partir de los valores centrales de tabla (a excepción del xenón, que es incorporado de forma aislada) sin perturbar. Esto significa que es necesario utilizar únicamente el valor de $\Sigma_x(Q, \mathbf{P}_g, \mathbf{P}_g) = \Sigma_x(Q, q', z, \theta_i = 1)$, obtenido a partir de una interpolación lineal de $n = 7$ parámetros que toma en cuenta todas las posibles combinaciones entre ellos. El xenón estacionario es incorporado mediante la derivada en la dirección del parámetro local, debido a que la derivada en la dirección del parámetro global es despreciable (ver ecuación 4), tal como puede verse en la figura 8. La siguiente ecuación resume la forma de computar las secciones eficaces estacionarias:

$$\Sigma_x^{\text{est}}(Q, \mathbf{P}_g^{\text{est}}, \mathbf{P}_g^{\text{est}}) \approx \Sigma_x(Q, q'_{\text{est}}, z_{\text{est}}, \theta_i^{\text{est}} = 1) + \left. \frac{\partial \Sigma_x}{\partial X e_l} \right|_{(Q, q'_{\text{est}}, z_{\text{est}})} \cdot (X e^{\text{est}} - X e^0). \quad (7)$$

En un transitorio, los valores de θ_i no equivalen necesariamente a 1. Por otro lado, las dependencias sobre el xenón y las temperaturas y densidades del refrigerante y moderador son tenidas en cuenta a partir de derivadas locales, tal como se muestra en el segundo término de la ecuación 5. Estas consideraciones resultan en la siguiente aproximación para la determinación de las secciones eficaces transitorias:

$$\Sigma_x^{\text{tr}}(Q, \mathbf{P}_g^{\text{est}}, \mathbf{P}_1^{\text{tr}}) \approx \Sigma_x(Q, q'_{\text{est}}, z_{\text{est}}, \theta_1^{\text{tr}}, \theta_2^{\text{tr}}, \theta_3^{\text{tr}}, \theta_4^{\text{tr}}) + \left. \frac{\partial \Sigma_x}{\partial X e_l} \right|_{(Q, q'_{\text{est}}, z_{\text{est}})} \cdot (X e_l^{\text{tr}} - X e^0) + \sum_{k=1}^{N_P} \left. \frac{\partial \Sigma_x}{\partial P_{l,k}} \right|_{(Q, q'_{\text{est}}, z_{\text{est}})} \cdot (P_{l,k}^{\text{tr}} - P_{l,k}^{\text{est}}). \quad (8)$$

En la siguiente sección se detalla la forma en que se determinan el valor central de tabla, las constantes del xenón y yodo y las derivadas con respecto a los parámetros previamente mencionados a partir de entradas de DRAGON y códigos y *scripts* que post procesan los resultados.

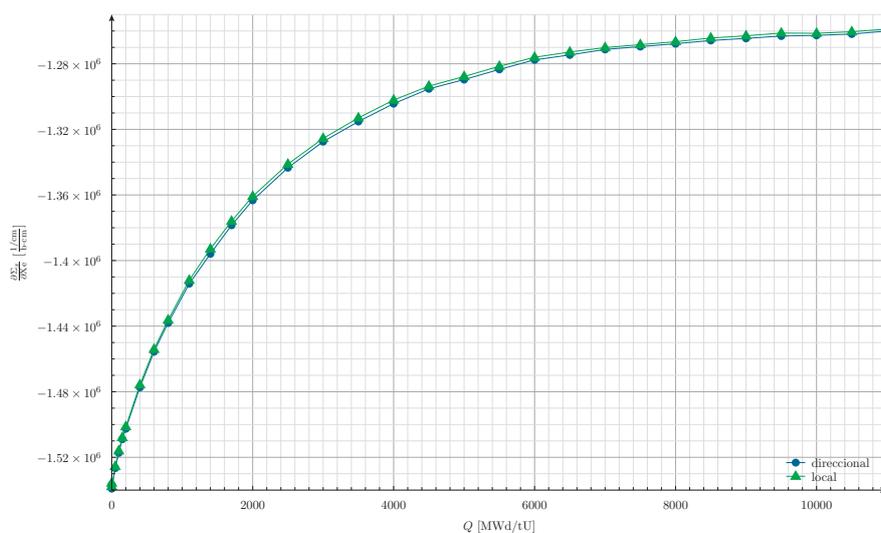


Figura 8: Derivada en la dirección del xenón local y en la dirección de $Xe_g = Xe_l$ (direccional) de la sección eficaz en función del quemado. El hecho de que ambas derivadas sean prácticamente iguales refleja que la derivada en la dirección del xenón global es despreciable. Si bien estas curvas corresponden a la sección eficaz de absorciones térmicas, lo mismo ocurre para el resto de las secciones eficaces.

6.2. Generación

Conceptualmente, la generación de la tabla de secciones eficaces requiere de la utilización de todos los temas abordados previamente. Estas tareas se han automatizado dado que se requiere de la producción y análisis de una gran cantidad de datos. En esta sección se describe la forma de elaborar la tabla a emplear en el modelo detallado de la planta a partir de una serie de *scripts* que ejecutan diferentes códigos e instrucciones.

La tabla 1 muestra qué valores se precisan obtener, por lo que se detallará la forma de calcular la información necesaria para determinar tanto los valores centrales de tabla Σ_x , las constantes del xenón y yodo (C_{Xe} y C_I) y las derivadas locales de las secciones eficaces respecto al parámetro P_k ($\partial\Sigma_x/\partial P$) $_{l,k}$ a partir de un único *script* descrito en el algoritmo 11.

En primer lugar, la llamada al *script* `run.sh` sin argumentos devolverá el *usage* del mismo, esto es:

```
$ ./run.sh

usage: ./run.sh <library_name> <tipo_de_calculo> [<parametro_th>]
donde library_name es el nombre de la librería a utilizar y ubicada dentro del directorio libraries,
y tipo_de_calculo corresponde a nested-loop, xenon-dynamics, xenon-derivs o pth-derivs.
En el último caso, el parametro_th equivalente a trefr, drefer, tmod, o dmod debe ser especificado.
Obs: las combinaciones sobre los parámetros globales son definidas en header.sh
```

Una vez especificada la biblioteca y definidas las combinaciones sobre los parámetros globales en el archivo `header.sh` es necesario especificar el tipo de cálculo. Para obtener los valores centrales de tabla, se ejecuta un tipo de cálculo `nested-loop`:

```
$ ./run.sh IAEA-E6 nested-loop
```

Estas ejecuciones serán paralelizadas y las salidas almacenadas en el directorio `outputs` bajo un nombre específico. Por otro lado, para obtener los datos necesarios para las C_{Xe} y C_I se realiza:

```
$ ./run.sh IAEA-E6 xenon-dynamics
```

Por último, los datos necesarios para obtener posteriormente las derivadas (mediante el *script* `multitable.sh` detallado en el algoritmo 12) respecto al xenón o parámetros termohidráulicos se realiza, respectivamente:

```
$ ./run.sh IAEA-E6 xenon-derivs
```

```
$ ./run.sh IAEA-E6 pth-derivs trefr | drefr | tmod | dmod
```

Obtenidos todos los objetos `multicompo` almacenados en el directorio `outputs` bajo nombres específicos, se procede a realizar la tabla de secciones eficaces en el subdirectorio `multitable` mediante:

```
$ ./multitable.sh
```

Algoritmo 11: *Script* `run.sh` para realizar las ejecuciones de DRAGON y así obtener los datos necesarios.

Datos: Argumentos del *script*.

Resultado: Directorios conteniendo las salidas de DRAGON.

si no se proveyeron argumentos entonces

└ imprimir el *usage* y abortar.

verificar argumentos provistos;

leer `header.sh` y definir combinaciones a realizar;

definir número de procesos a ejecutar en paralelo;

definir parámetros comunes a todos los procesos a ejecutar;

para cada altura de núcleo

└ calcular parámetros termohidráulicos globales;

└ **para cada potencia lineal**

└└ calcular perfil de temperatura del combustible;

└└ condensar perfil a volúmenes de DRAGON;

└└ especificar nombre de directorio de corrida dependiendo del tipo de cálculo y parámetros globales;

└└ **si el tipo de cálculo requiere de perturbaciones locales entonces**

└└└ definir perturbaciones locales dependiendo cual sea el parámetro;

└└└ reemplazar perturbaciones locales en el *template* del *input* que corresponda;

└└ reemplazar parámetros globales en el *template* del procedimiento `DefParams`;

└└ mover al directorio correspondiente el *input* y los procedimientos;

└└ ejecutar DRAGON;

└└ **si se alcanzó el número de procesos a paralelizar entonces**

└└└ esperar la finalización de los procesos;

/* si el # de cálculos no es múltiplo del # de procesos a ejecutar en paralelo */

si quedaron procesos lanzados entonces

└ esperar la finalización de los procesos;

Algoritmo 12: *Script multitable.sh* para obtener tabla de secciones eficaces.

Datos: Objetos multicompo.

Resultado: Tabla de secciones eficaces y sus derivadas.

leer header .sh y definir combinaciones a realizar;

verificar la existencia de todos los objetos multicompo necesarios;

para cada altura de núcleo

 para cada potencia lineal

 compilar rutinas de extracción de datos;

 generar tabla parcial de valores centrales de tabla;

 generar tabla parcial de constantes para la dinámica del yodo y xenón;

 generar tabla parcial de derivadas respecto a la concentración de xenón;

 generar tabla parcial de derivadas respecto a la temperatura de refrigerante;

 generar tabla parcial de derivadas respecto a la densidad de refrigerante;

 generar tabla parcial de derivadas respecto a la temperatura de moderador;

 generar tabla parcial de derivadas respecto a la densidad de moderador;

 unir tablas parciales para formar una 'matriz fila' de la tabla final;

 agregar 'matriz fila' a la tabla final;

 limpiar datos y archivos generados;

separar tabla final en secciones eficaces y sus derivadas;

Referencias

- [1] A. Hébert G. Marleau y R. Roy. *A USER GUIDE FOR DRAGON VERSION5*. Inglés. Inf. téc. IGE-335. Institut de génie nucléaire, Département de génie mécanique, École Polytechnique de Montréal, 2015.
- [2] A. Hébert y R. Roy. *THE GANLIB5 KERNEL GUIDE*. Inglés. Inf. téc. IGE-332. Institut de génie nucléaire, Département de génie mécanique, École Polytechnique de Montréal, 2016.
- [3] G. Theler. *Description of the computational tool wasora*. Inglés. Inf. téc. CIT-WSWA-TD-9E3D. 2016.
- [4] J. P. Gómez Omil. *Description of the plug-in xdfrrpf*. Inglés. Inf. téc. 10661-N-IT16-001. TECNA S.A., 2016.
- [5] Information Systems Laboratories Inc. *RELAP5/MOD3.3 Code Manual Volume I: Code Structure, System Models, and Solution Methods*. Inf. téc. NUREG/CR-5535/Rev 1. 2001.
- [6] G. Marleau. *New Geometries Processing in DRAGON: The NXT: Module*. Inglés. Inf. téc. IGE-260. Institut de génie nucléaire, Département de génie physique, École Polytechnique de Montréal, 201.
- [7] D. G. Cacuci. *Handbook of Nuclear Engineering*. Inglés. Springer, 2010.
- [8] G. Theler y M. Rivero. *PUMITACPL: Extensión del Programa Cinético Espacial de cálculo neutrónico 3D para permitir simulaciones con realimentación termohidráulica y señales de control externas*. Inf. téc. 10105-G-IT-004. TECNA S.A., 2010.
- [9] G. Giuntoli. *Modelos de RELAP para la determinación de perfiles de temperatura del combustible*. Inf. téc. 10527-N-IT16-316. TECNA S.A., 2016.