

PUMITA

CARLOS GRANT, RUBERT MARTÍN



Manuales de uso y soporte

Mayo 2018 – version 1.0

Carlos Grant, Rubert Martín: *PUMITA*, Manuales de uso y soporte, ©
Mayo 2018

SUPERVISORES:

Alexis Weir, Ariel Hosid

GRUPO:

Grupo de Desarrollo de Códigos Nucleares

DEPARTAMENTO:

Física de Reactores Avanzados, Centro Atómico Bariloche

ÍNDICE GENERAL

I MANUAL DE USUARIO

1	QUÉ ES PUMITA	2
1.1	Motivación del surgimiento de pumita	2
1.2	Objetivos de Pumita	2
1.3	Base de desarrollo	2
2	CAPACIDADES DE PUMITA	3
3	DESCRIPCIÓN DEL MODELO DE NÚCLEO	4
3.1	Modelo geométrico	4
3.2	Materiales	5
3.3	Representación de las barras de control	8
4	INTERFAZ DE PUMITA PARA USO EXTERNO	10
4.1	Incluyendo la librería de pumita en el programa externo	11
5	CÓMO HACER SIMULACIONES CON pumita	13
5.1	Archivo de entrada ENTRADA.TXT	13
5.1.1	Variables de condiciones de cálculo numérico en ENTRADA.TXT	13
5.1.2	Variables de estado inicial en ENTRADA.TXT	16
5.2	Estado inicial del reactor	17
5.2.1	Guardar y cargar estado	18
5.3	Avanzar un paso de simulación	18
6	POST-PROCESAMIENTO DE RESULTADOS	20
6.1	Post-procesamiento externo	20
6.2	Archivo SALIDA.TXT	20
6.2.1	Encabezado	20
6.2.2	Opciones de impresión	20
6.3	pumita2vtk	21
6.3.1	Formato PUMA	22
6.3.2	Empleando el utilitario puma2vtk	23

II MANUAL DEL PROGRAMADOR

7	MÓDULOS DE LA DLL	25
7.1	Módulo reactor.PAS	25
7.2	Módulo CAREMTRI.PAS(y su versión de geometría hexagonal CAREMHEXA.PAS)	25
7.3	Módulo interpol.PAS	25
8	INICIALIZACIÓN, VARIABLES GLOBALES	26
9	CÁLCULO ESTÁTICO	27
9.1	Procedimiento UnPaso.	27
9.2	Procedimiento StaticCalculation	27
10	CÁLCULO DINÁMICO	35

10.1	Procedimiento AdiabaticMethod	35
10.2	Procedimiento DirectMethod	36
11	CÁLCULO DE PARÁMETROS NEUTRÓNICO-TERMOHIDRÁULICOS	38
11.1	Secciones eficaces	38
11.2	Variables <i>switch</i>	39
11.3	Valores centrales	39
11.4	Cálculo estático	40
11.5	Cálculo estático de Xenón	42
11.5.1	Variables empleadas en el código	42
11.5.2	Cálculo del Xenón	43
11.5.3	Cálculo estático del Samario y de los Parámetros termohidráulicos	44

III APÉNDICES

A	DESCRIPCIÓN DEL FORMATO DE LAS TABLAS DE SECCIONES EFICACES	46
B	DESCRIPCIÓN DEL ARCHIVO DEL ESTADO INICIAL DEL REACTOR	48

	BIBLIOGRAFÍA	50
--	--------------	----

ÍNDICE DE FIGURAS

Figura 3.1	Numeración de los canales del modelo del núcleo en simetrías 1 y 3	4
Figura 3.2	Denominación alfanumérica de los elementos combustibles.	5
Figura 3.3	Red Geométrica para el cálculo neutrónico	5
Figura 3.4	Distribución de materiales en el núcleo	7
Figura 3.5	Ubicación de las barras de control	9
Figura 9.1	Diagrama de Flujo para el procedimiento StaticCalculation	28
Figura 9.2	Diagrama de flujo para el procedimiento InterpolateXS	30
Figura 9.3	Diagrama de flujo para el procedimiento Iteraciones	32
Figura 9.4	Diagrama de flujo para el procedimiento Potencias	34

ÍNDICE DE TABLAS

Cuadro 3.1	Tablas de materiales para el cálculo de secciones eficaces empleadas en el modelo de PUMITA	6
Cuadro 3.2	Valores centrales de temperatura y densidad a los que corresponden los juegos de secciones eficaces de cada material listado en la Tabla 3.1, para cada estado operativo del reactor.	7
Cuadro 3.3	Valores centrales de temperatura y densidad a los que corresponden los juegos de secciones eficaces obtenidos para el agua del reflector, para cada estado operativo del reactor.	8
Cuadro 3.4	Composición de los sistemas de Extinción Rápida (SER! (SER!)), Ajuste y Control (Sistema de Ajuste y Control (SAC)) y resguardo, mostrando sus correspondencias con los números de bancos de control y los elementos combustibles que los componen	8
Cuadro 11.1	Valores centrales de parámetros para los cuales están definidas las secciones eficaces tabuladas	40

ABREVIATURAS

SAC	Sistema de Ajuste y Control
SER	Sistema de Extinción Rápida
VQ	Venenos Quemables

Parte I

MANUAL DE USUARIO

QUÉ ES PUMITA

PUMITA es un módulo de enlace dinámico (DLL) para Windows que calcula el flujo neutrónico y otros parámetros de interés del núcleo del reactor CAREM.

1.1 MOTIVACIÓN DEL SURGIMIENTO DE PUMITA

PUMITA surge como resultado de un requerimiento de trabajo [2] de Control de Procesos y DIFRA al Departamento Estudio de Reactores y Centrales, Gerencia Reactores y Centrales Nucleares, de la Gerencia de Área Energía Nuclear.

Este requerimiento parte de la necesidad de obtener un modelo de núcleo detallado a través del acople RELAP5/PUMITA, que permita representar con mayor exactitud los resultados de las simulaciones con los Simuladores de Desarrollo y Full Scope, especialmente aquellas donde se producen cambios significativos en la distribución de potencia.

1.2 OBJETIVOS DE PUMITA

PUMITA está pensado para ser acoplado a modelos de planta y termohidráulicos especializados en simuladores de tiempo real.

1.3 BASE DE DESARROLLO

PUMITA fue originalmente programado en el entorno de desarrollo DELPHI con el lenguaje de programación Object Pascal. También se le hicieron modificaciones posteriores en el entorno de desarrollo Lazarus para el mismo lenguaje de programación.

Existe actualmente una propuesta en el Grupo de Desarrollo de Códigos Nucleares de pasar el código original a un lenguaje con más soporte actual como FORTRAN ó C++.[xxx referencia a la tesis]

El código fuente está basado en el código neutrónico para núcleo PUMA. El propio código incluye características específicas del reactor CAREM, a fin de hacer más rápido el cálculo con el mismo y así cumplir su finalidad de uso en un simulador de tiempo real.

CAPACIDADES DE PUMITA

PUMITA cuenta con las siguientes capacidades:

1. Calcular en geometría triangular (6 triángulos por elemento combustible) o hexagonal (1 hexágono por elemento combustible) estados estacionarios y transitorios para diferentes modelos 3D del núcleo del reactor CAREM en tiempo real, realizando un cálculo de cinética espacial.
2. Representar el núcleo del reactor CAREM para diferentes estados de quemado. Para cambiar los estados de quemado de núcleo, se cambian solamente los archivos de entrada al código sin tener que modificar el código fuente ni compilarlo nuevamente.
3. Modelar la inserción de todas las barras de control (SAC y SER) con pasos de cualquier medida. En el modelo ya están incluidas las barras con un tratamiento del efecto cúspide[xx insertar referencia al efecto cúspide xxxx].
4. PUMITA modela el flujo neutrónico con 5 grupos de energía. Actualmente se están haciendo estudios para flexibilizar la cantidad de grupos energéticos[xxx referencia a la tesis xxx].
5. PUMITA calcula concentraciones de precursores y venenos (Xenón y Samario) y su evolución temporal para diferentes transitorios. Las distribuciones de ambos isótopos y los asociados con ellos se deberán calcularse en estado estacionario junto con todos los otros parámetros neutrónicos y termohidráulicos del reactor al comienzo de la simulación. Se podrá tener información separada de los valores de las concentraciones de los mismos en todo instante de la simulación.
6. PUMITA modela el calor de decaimiento durante la operación y para el reactor apagado.
7. PUMITA cuenta con parámetros que le permiten configurar el tiempo total y el paso de tiempo de simulación tanto para el cálculo de estados estacionarios como para transitorios.

3.1 MODELO GEOMÉTRICO

PUMITA trabaja con una geometría fija equivalente a las que se usan en CITVAP y PUMA que se describirá a continuación.

En la Fig. 3.1 puede verse la numeración en simetría 1 utilizada por PUMA y la numeración en simetría 3 utilizada por CITVAP. En la Fig. 3.2 se muestra la codificación alfanumérica de los canales.

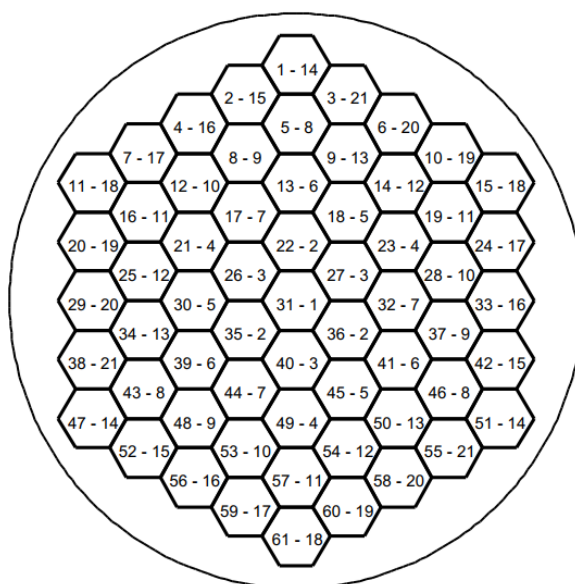


Figura 3.1: Numeración de los canales del modelo del núcleo en simetrías 1 y 3

Para la modelización geométrica en el módulo, tanto en PUMA como en CITVAP, se empleó una red que representa a cada elemento combustible con 6 triángulos equiláteros de 4 cm de altura, tal como puede verse en la Fig. 3.3 ; el paso de red entre elementos es de 16 cm.

Los trozos axiales en la zona del núcleo coinciden con los segmentos de la red de cálculo y se numeran de 1 a 14 comenzando desde la parte inferior. Los separadores se encuentran en los trozos 6 y 11.

El círculo interno de los dos que rodean al núcleo, con un radio de 78 cm (Fig. 3.3), marca el límite interno del barrel. Se impone provisoriamente la condición de corriente entrante nula en la parte externa de los triángulos que rodean al núcleo.

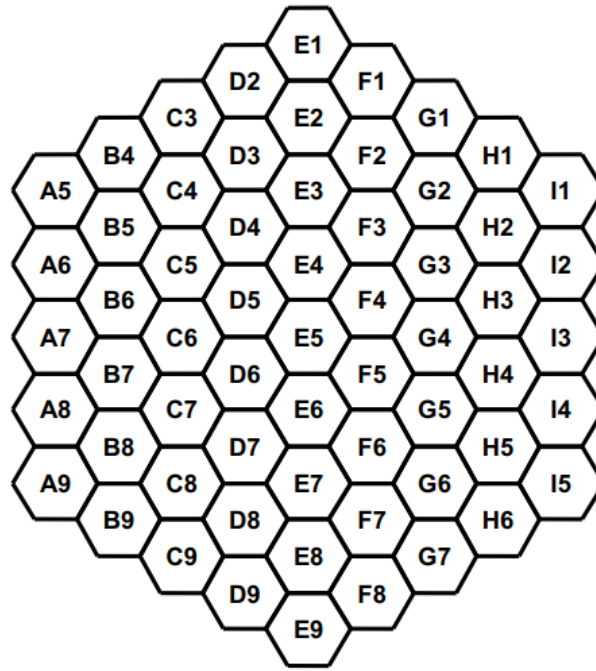


Figura 3.2: Denominación alfanumérica de los elementos combustibles.

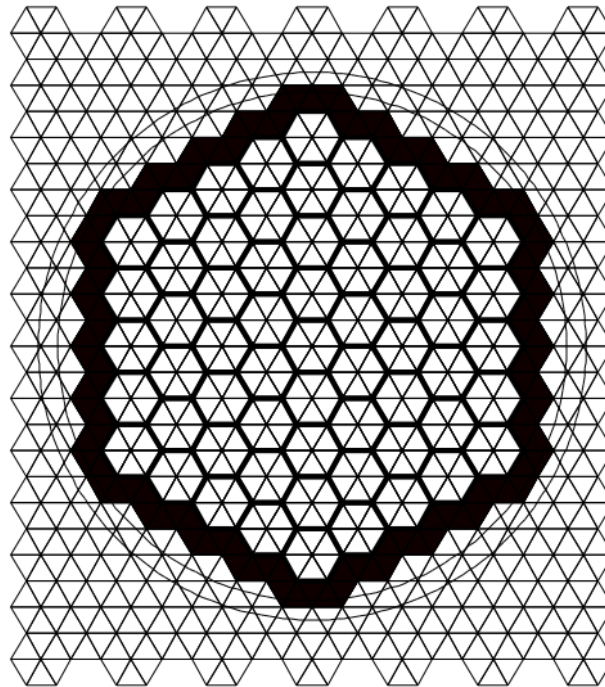


Figura 3.3: Red Geométrica para el cálculo neutrónico

3.2 MATERIALES

Los materiales son asignados mediante tablas de secciones eficaces a 5 grupos de energía en ambos sistemas. Los datos de tablas de

No. de tabla	Enriquecimiento [%]	Cantidad de barras con VQ	Cantidad de barras absorbentes	Presencia de separadores
1	1,8	0	0	No
2	3,1	0	0	No
3	3,1	6	0	No
4	3,1	12	0	No
5	1,8	0	12	No
6	3,1	0	12	No
7	3,1	0	18	No
8	3,1	6	12	No
9	3,1	6	18	No
10	1,8	0	0	Si
11	3,1	0	0	Si
12	3,1	6	0	Si
13	3,1	12	0	Si
14	1,8	0	12	Si
15	3,1	0	12	Si
16	3,1	0	18	Si
17	3,1	6	12	Si
18	3,1	6	18	Si

Tabla 3.1: Tablas de materiales para el cálculo de secciones eficaces empleadas en el modelo de PUMITA

materiales se prepararon para el caso de reflector de agua se prepararon 18 tablas de secciones eficaces. Cada una de ellas aplicable a cualquiera de las partes del reactor en las que se da la combinación que determina cada tabla en particular, entendiendo por tal combinación la del porcentaje de enriquecimiento, la cantidad de barras con VQ (6 ó 12), y la cantidad de barras absorbentes de los elementos de control (12 ó 18). Además, en la distribución de materiales se tiene en cuenta el hecho de que los elementos combustibles pueden contener un separador.

La distribución radial de los materiales según su enriquecimiento, VQ y barras absorbentes puede verse en las Figs. 3.4 y 3.5. La distribución de materiales para cada conjunto (bloque) de secciones eficaces se detalla en la Tabla 3.1. Las tablas 10 a 18 corresponden a los mismos materiales que los de las 1 a 9 pero con los separadores incluidos en la desmoralización de la celda.

Se generaron tablas de secciones eficaces para el núcleo isotérmico a 20°C, 257°C y 450°C correspondientes a los estados operativos de

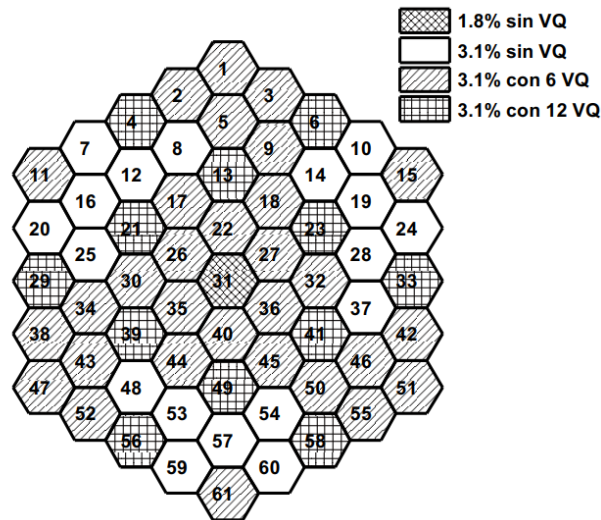


Figura 3.4: Distribución de materiales en el núcleo

Temperatura del combustible [°C]	Temperatura del refrigerante [°C]	Densidad del refrigerante [g/cm ³]
450	305	0,6750

Tabla 3.2: Valores centrales de temperatura y densidad a los que corresponden los juegos de secciones eficaces de cada material listado en la Tabla 3.1, para cada estado operativo del reactor.

parada fría, crítico caliente y operación normal (véase [3] y [4]). Para el módulo presentado en este trabajo interesa solamente los valores a operación normal.

Los valores centrales de temperatura y densidad a los que corresponden los juegos de secciones eficaces de cada material listado en la Tabla 3.1, son los consignados en la Tabla 3.2. En la Tabla 3.3 se consignan los valores de los parámetros termohidráulicos para el agua del reflector.

Para el desarrollo de PUMITA se simplificó algo la interpolación en las tablas y la dependencia de las secciones eficaces respecto a los parámetros termohidráulicos, tratándose con más detalle la dependencia respecto a la densidad del refrigerante y suponiendo total linealidad para la temperatura del combustible y la del refrigerante.

Una descripción del formato de las tablas puede verse en el ANEXO A.

La distribución axial de los elementos combustibles con VQ no es homogénea. Para el caso de estos elementos combustibles los Venenos Quemables (VQ) se incluyen entre los trozos 2 y 12 (véase [3]) los trozos 1, 13 y 14 nunca tienen VQ (Véase [4]).

Reflector Inferior		Reflector Radial		Reflector Superior	
Temperatura [°C]	Densidad [g/cm ³]	Temperatura [°C]	Densidad [g/cm ³]	Temperatura [°C]	Densidad [g/cm ³]
284	0.7530	308	0.7020	326	0.6500

Tabla 3.3: Valores centrales de temperatura y densidad a los que corresponden los juegos de secciones eficaces obtenidos para el agua del reflector, para cada estado operativo del reactor.

Banco	Elementos combustibles	Función
1 (central)	E5	Control fino de reactividad (SAC)
2	E4 – D6 – F5	SAC
3	F4 – D5 – E6	SAC
9	D3 – C8 – H4	SAC
10	H3 – C4 – D8	SER!
11	H2 – B5 – E8	SAC
12	G2 – B6 – F7	SER!
8	D4 – D7 – G4	SER!
13	F2 – B7 – G6	SAC
7	E2 – B8 – H5	Banco de resguardo

Tabla 3.4: Composición de los sistemas de Extinción Rápida (SER!), Ajuste y Control (SAC) y resguardo, mostrando sus correspondencias con los números de bancos de control y los elementos combustibles que los componen

3.3 REPRESENTACIÓN DE LAS BARRAS DE CONTROL

Las barras de control son representadas en PUMA dando directamente las secciones eficaces del elemento con la barra de control insertada, asignándole al trozo axial correspondiente el número de material que lo caracteriza.

Los bancos del SAC, del SER! y de resguardo, reciben una denominación en correspondencia con la numeración en simetría 3, de los elementos combustibles que los integran. En la Tabla 3.4 se detalla la composición de estos sistemas (véanse asimismo Figs. 3.2 y 3.5).

El orden de inserción de los bancos en la búsqueda de criticidad es 1, 2, 9, 11, 13 y 3 y el de extracción es el inverso (véase [3]).

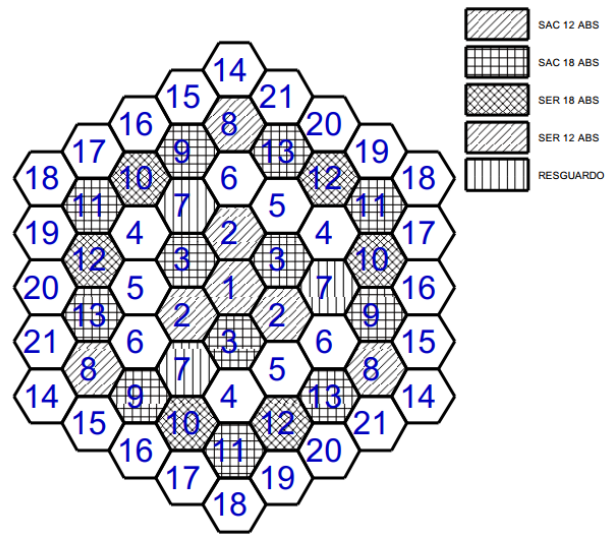


Figura 3.5: Ubicación de las barras de control

INTERFAZ DE PUMITA PARA USO EXTERNO

PUMITA cuenta con una interfaz para intercambio de datos con el modelo externo al mismo. Esta interfaz está dada por las siguientes constantes, tipos, procedimientos y funciones de Pascal:

```

const
    NTROZOS = 14;
    NCANALES = 61;
    NGrupos = 5;

type
    GroupType = array[1..NGrupos] of real;
    CoreMatrix = array[1..NCANALES,1..NTROZOS] of real;
    CoreFluxMatrix = array[1..NCANALES,1..NTROZOS] of
        GroupType;
    ChannelType = array[1..NCANALES] of real;
    MemType = array[0..2500000] of byte;

procedure InsercionBancos
    (var B01,B02,B09,B11,B13,B03,B08,B10,B12,B07: real);
// Inserta los bancos B01, B02, etc., en las posiciones dadas.

procedure InsercionBarras
    (var E5, E4,D6,F5, D3,C8,H4, H2,B5,E8, F2,B7,G6, F4,D5,E6,
    E2,B8,H5, H3,C4,D8, G2,B6,F7, D4,D7,G4: real);
// Inserta las barras E5, E4, E6, D6, F5, D3, etc.,
// en las posiciones dadas.

procedure PideBarras
    var E5, E4,D6,F5, D3,C8,H4, H2,B5,E8, F2,B7,G6, F4,D5,E6,
    E2,B8,H5, H3,C4,D8, G2,B6,F7, D4,D7,G4: real);
// Pide las inserciones de barra utilizadas a Pumita

procedure UnPaso(var T,DT: real);
// Pide avanzar un paso de DT segundos en la simulacion
// del reactor por medio de la cinetica
// espacial. Si se da DT = 0.0 se supone que se pide
// hacer un calculo en estado estacionario.

procedure pedir_parametros_puntuales
    (var PotTotal0, PotResidual0, R00, KE0,
    DerivadaLogaritmica0, FForma0, PMax0: real;
    var CanalPMax0, TrozoPMax0: integer;
    var PotMax0: real; var CanalPotMax0: integer;
    var ConcXEMedia0, ConcXenon00: real;
    var ConcSMMedia0, ConcSamario00: real;
    var NumIter: Integer; var EPS,EPSKE: real) ;
// Pide los valores de la potencia total, la de decaimiento,
```



```

// la reactividad, el K efectivo inicial, la derivada
// logaritmica, el factor de forma, la potencia maxima
// y su localizacion, la potencia maxima por canal y su
// localizacion, la concentracion media de xenon y samario
// y sus concentraciones iniciales, el numero de iteraciones,
// la precision en el flujo y el K efectivo.

procedure pasar_parametros_TH(var TC, TR, DR: CoreMatrix) ;
// Le pasa a PUMITA la distribucion tridimensional
// de la temperatura del combustible, la del
// refrigerante y la densidad del refrigerante.

procedure pasar_BORO(var Boro: CoreMatrix) ;
// Le pasa a PUMITA la distribucion tridimensional de boro.

procedure pedir_distribucion_potencias
(var P: CoreMatrix; var PC: ChannelType) ;
// Le pide a PUMITA la distribucion de potencia por canal
// y trozo y la potencia total por canal

procedure pedir_flujos(var F: CoreFluxMatrix) ;
// Le pide a PUMITA la distribucion de flujos a 5 grupos.

procedure Transferir(var TTT: real;
var grabar: integer; var buffer: MemType);
// Le pide a PUMITA que grabe su estado en un
// archivo de nombre StateFile (var = 1) o que lea (var = 0).

procedure pedir_parametros_TH(var TC, TR, DR: CoreMatrix) ;
// Le pide a PUMITA la distribucion tridimensional
// de la temperatura del combustible, la del
// refrigerante y la densidad del refrigerante.

```

4.1 INCLUYENDO LA LIBRERÍA DE PUMITA EN EL PROGRAMA EXTERNO

La forma de cargar la librería en el modulo externo dependerá de las especificaciones del mismo.

A modo de ejemplo, si un programa hecho en C++ requiere las funciones de la librería, se deberá incluir la línea de código siguiente:

```

..
#include <Windows.h>
..
int main()
{
    ..
    // cargar la libreria
    HMODULE dll = LoadLibrary('reactor_fab_x64')
    ..
    //asignar funciones de la libreria a funciones locales de
    C++

```

```
    DLL_FUNC1 func1 = (DLL_FUNC1) GetProcAddress( hDLL, "
        name_of_func1" );
    ..
    x=func1( 1, 2 );
    ..
}
```

Los detalles sobre incorporación de librerías externas y el uso de las funciones ahí descritas quedarán a cuenta de la documentación del programa externo.

PUMITA recibe del sistema externo las distribuciones de parámetros termohidráulicos tales como Temperatura del Combustible, Temperatura del Refrigerante, Densidad del Refrigerante, Concentración de Boro, Posición de cada barra de Control y Paso de Tiempo, mediante llamadas a procedimientos debidamente programados. Esto se realiza mediante los procedimientos descritos en el punto anterior.

Si se solicita un cálculo en estado estacionario o una evolución en el paso del tiempo estipulado devuelve la distribución tridimensional de potencia y flujo neutrónico y de otros parámetros globales tales como Reactividad, Factor de Forma, Potencia Máxima, etc). Los procedimientos `pedir_potencias`, `pedir_flujos` y `pedir_parametros_puntuales` cumplen estas funciones.

CÓMO HACER SIMULACIONES CON PUMITA

Para realizar simulaciones con PUMITA del comportamiento del CA-REM, deberán establecerse las condiciones iniciales del reactor y se realizarán tantas llamadas al procedimiento UnPaso como pasos de tiempo se quieran tener dentro de la simulación

5.1 ARCHIVO DE ENTRADA `entrada.txt`

El archivo ENTRADA.TXT permite definir los valores iniciales de variables que definen el estado inicial del reactor. Además, se especifican condiciones para el cálculo numérico. Un ejemplo de archivo de entrada puede ser encontrado en el Apéndice [B](#)

5.1.1 *Variables de condiciones de cálculo numérico en ENTRADA.TXT*

5.1.1.1 OPCIONES

Para especificar las opciones del cálculo numérico, debe de escribirse en el archivo ENTRADA.TXT una línea que empiece con la palabra clave OPCIONES seguida de una o más de las siguientes palabras claves:

- XENON: Para activar la variable switch (véase la sección [11.2](#)) ConXenon (cambiar su valor a TRUE) y por tanto habilitar el cálculo de la concentración de Xenón y su influencia en las secciones eficaces.
- SAMARIO: Para activar la variable switch SAMARIO (cambiar su valor a TRUE) y por tanto habilitar el cálculo de la concentración de Samario y su influencia en las secciones eficaces.
- REACOPLAMIENTO TERMOHIDRAULICO: Para activar la variable switch ConReacoplamientoTermohidraulico (cambiar su valor a TRUE) y por tanto habilitar la influencia de las variables termohidráulicas (Temperatura y Densidad del Refrigerante, Temperatura del Combustible) en las secciones eficaces.
- CINETICA: Se especifica a PUMITA que se pretende (además del cálculo estático inicial) realizar un cálculo dinámico.
- DIRECTO: Se especifica a PUMITA que se desea realizar un cálculo directo (Debe de estar también la opción CINETICA).

- **ADIABATICO**: Se especifica a PUMITA que se desea realizar un cálculo adiabático (Debe de estar también la opción **CINETICA**). En realidad, si ninguna de las opciones **ADIABATICO** o **DIRECTO** se especifica, PUMITA supone por defecto que se realizará un cálculo adiabático.

Para una explicación de los métodos adiabático y directo véase la referencia [1] y el capítulo 10.

5.1.1.2 ESTADO

Si se desea guardar o leer un estado en (o desde) un archivo externo, para simulaciones posteriores, se deberán especificar los siguientes comandos en ENTRADA.TXT:

- Guardar estado:

```
GUARDAR ESTADO
nombre_del_archivo_externo
```

- Leer estado:

```
Leer ESTADO
nombre\_del\_archivo\_externo
```

5.1.1.3 Matriz de respuesta

Se especificará la opción

```
MATRIZ RESPUESTA
```

en una línea independiente de ENTRADA.TXT para activar el cálculo con Matriz de Respuesta en los bordes del núcleo.

5.1.1.4 Número máximo de iteraciones

Para los cálculos estático y directo se puede especificar un número máximo de iteraciones con la línea:

```
MAXIMO ITERACIONES
No_it
```

O sea, primero la directiva, y en la siguiente línea, el número máximo de iteraciones.

5.1.1.5 Opciones de impresión

Véase la subsección 6.2.2.

5.1.1.6 Coeficiente de Sobrerrelajación

Este importante parámetro permite acelerar las iteraciones para la convergencia del flujo neutrónico durante el cálculo con la ecuación de difusión. No hay una regla universal para elegir el valor óptimo

de este coeficiente (el que provoca el número mínimo de iteraciones para la convergencia), por lo que se recomienda hacer un estudio paramétrico inicial de la cantidad de iteraciones como función del coeficiente, para cada transitorio que se quiera estudiar. La influencia de este coeficiente no depende del paso de tiempo empleado.

Para especificar el coeficiente de sobrerrelajación en el archivo ENTRADA.TXT se especifican las líneas:

```
COEFICIENTE SOBRERRELAJACION
No_coef
```

5.1.1.7 Control de tiempos de simulación

La variable paso de tiempo (DT) se puede modificar en el archivo ENTRADA.TXT con el comando:

```
PASO TIEMPO
No\_DT
```

5.1.1.8 Precisión

Para controlar el cálculo iterativo se pueden pasar los valores de precisión para el cálculo estático y el dinámico por separado:

Para el cálculo estático:

```
PRECISION
No_precEstatico
```

Para el cálculo dinámico:

```
PRECISION CINETICO
No_precCinetico
```

5.1.1.9 Interpolación

Para especificar que se desea utilizar la interpolación lineal en las tablas de secciones eficaces se indicará en ENTRADA.TXT:

```
INTERPOLACION LINEAL
```

Para indicar que se desea emplear interpolación múltiple:

```
INTERPOLACION MULTIPLE
archivo_interpolacion_multiple
```

o sea, se deberá pasar en una segunda línea el archivo de interpolación múltiple.

5.1.1.10 Potencia por fisión

Se puede indicar la potencia por fisión con el archivo ENTRADA.TXT usando la directiva

```
POTENCIA FISION
No_PPorFision
```

5.1.2 Variables de estado inicial en ENTRADA.TXT

5.1.2.1 Archivo de tablas de secciones eficaces

Para especificar el archivo externo del cuál se leeran las tablas de secciones eficaces se especifica en ENTRADA.TXT la siguiente opción:

```
TABLAS
Nombre_del_archivo_de_tablas
```

5.1.2.2 Valores centrales de Tabla

Para realizar interpolaciones con los valores de las secciones eficaces incrementales de las Tablas, deben de especificarse valores centrales de tabla para los distintos parámetros neutrónico-termohidráulicos.

En el archivo ENTRADA.TXT esto se puede hacer con el comando:

```
TABLA VALORES CENTRALES
ConcXE0, TComb0, DensRefr0, TemprRefr0
```

5.1.2.3 Asignación de materiales

Se especifican los números de materiales (Tabla xx) por canal y trozo en el archivo ENTRADA.TXT de la forma:

```
MATERIALES
NumMaterial[canal=1,trozo=1] NumMaterial[canal=1,trozo=2] ....
    NumMaterial[canal=1,trozo=NTR0Z0S] ...
NumMaterial[canal=NCANALES,trozo=NTR0Z0S]
```

5.1.2.4 Temperatura de combustible, temperatura de refrigerante, Densidad de Refrigerante, Quemado

Siguiendo el mismo formato que el caso de los materiales (sección anterior), se pueden especificar valores iniciales para los parámetros Temperatura de combustible, temperatura de refrigerante, Densidad de Refrigerante y Quemado. Para ello se deberán usar acordeamente las siguientes palabras claves en el archivo ENTRADA.TXT.

```
TEMPPERATURA COMBUSTIBLE
TEMPERATURA REFRIGERANTE
DENSIDAD REFRIGERANTE
QUEMADO
```

Estas palabras claves deberán de estar seguidas por los correspondientes valores por canal y trozo, con el mismo formato especificado para MATERIALES en la sección anterior.

5.2 ESTADO INICIAL DEL REACTOR

Una vez establecidos los valores iniciales de las variables mencionadas en 5.1.2, se debe pasar a calcular el estado inicial convergido del reactor.

El reactor idealmente debe de partir de un estado crítico al iniciar la simulación. Para ello se deberán de realizar iteraciones de cálculo estático interactuando con el modelo externo termohidráulico hasta que los parámetros neutrónicos termohidráulicos se encuentren acoplados.

Esto requerirá una secuencia de transferencias de parámetros neutrónicos (en particular la potencia neutrónica por canal y trozo) al modelo termohidráulico externo; así como una transferencia desde este modelo de parámetros termohidráulicos (Temperaturas y densidades).

Cada iteración para lograr el acople estático inicial se realizará siguiendo el siguiente proceso iterativo:

1. Se llama al procedimiento `UnPaso` con parámetros $T=0.0$, $DT=0.0$, para invocar el cálculo estático del flujo y demás parámetros neutrónicos (potencias, concentraciones de isótopos, etc.).
2. Si el flujo neutrónico se encuentra convergido (esto se puede saber por la cantidad de iteraciones que llevó el cálculo del flujo neutrónico del paso anterior al actual), se detiene el proceso iterativo, de lo contrario...
3. Se llama al procedimiento `pedir_distribucion_potencias` para obtener potencia total por canal y trozo.
4. Se calculan los parámetros termohidráulicos (temperaturas de combustible y refrigerante, densidad de refrigerante) a ser pasados a PUMITA .
5. Se pasan los parámetros a PUMITA con el procedimiento `pasar_parametros_TH`, y se procede a reiniciar el proceso iterativo.

Debe de tenerse en cuenta que además de los parámetros termohidráulicos, existen otros parámetros que pueden influir en la convergencia del flujo neutrónico para el estado acoplado inicial. Estos son parámetros neutrónicos que el propio PUMITA calculará en cada llamada a `UnPaso` si las correspondientes variables switch están activas: las distribuciones de Xenón y de Samario.

Una guía explicativa paso a paso de la parte del acople neutrónico termohidráulico en la que interviene PUMITA puede ser encontrada en el capítulo 11.

5.2.1 Guardar y cargar estado

Una vez que se alcanzó el estado inicial crítico y acoplado, resulta conveniente guardarlo para posteriores simulaciones. Para ello está el procedimiento `Transferir`. El mismo realiza una copia en un buffer de memoria de todo el estado del reactor, incluyendo las variables de entrada y salida, y entrega dicho buffer. El contenido de este buffer puede ser utilizado posteriormente para cargar el estado.

La misma función `Transferir` permitirá cargar el estado completo a partir de un estado previamente salvado en un buffer de memoria incluyendo todos los valores de todas las variables en el momento de ser salvadas.

5.3 AVANZAR UN PASO DE SIMULACIÓN

Una vez establecido el estado inicial acoplado del reactor, se realizará la simulación mediante paulatinas llamadas al procedimiento `UnPaso`.

Este procedimiento realiza los cálculos de la simulación correspondientes al avance en un paso de tiempo y retorna el valor actual del tiempo.

Al presente no están implementados los códigos de error y sus mensajes descriptivos, pues su especificación deberá surgir de las experiencias realizadas en conjunto con el modelo termohidráulico.

En cada paso de tiempo PUMITA podrá entregar la distribución de potencia y el flujo neutrónico para todos los grupos de energía a lo largo de todos los nodos que conforman el núcleo mediante `pedir_flujos` y `pedir_distribucion_potencias`. Falta aún programar la estimación del flujo neutrónico en los puntos del reflector que se requieran, para lo cual el Proyecto proveerá la relación entre la indicación de los detectores externos al reflector y los flujos en el reactor en las zonas más cercanas a los mismos.

Así mismo, PUMITA utiliza información del refrigerante, moderador y combustible en cada nodo neutrónico (incluyendo los del reflector) para actualizar sus condiciones (procedimiento `pedir_parametros_TH`) según se indica en las siguientes sentencias:

1. Distribución de la temperatura del combustible (procedimiento `pedir_parametros_TH`).
2. Asociadas a volúmenes hidrodinámicos:
 - Distribución de la densidad del refrigerante (procedimiento `pedir_parametros_TH`)
 - Distribución de la temperatura del refrigerante (procedimiento `pedir_parametros_TH`).

- Distribución de la concentración de boro: la pasa el sistema de control por medio del procedimiento pasar_BORO.
- Posición de las barras de control: la pasa el sistema de control por medio del procedimiento insercion_barras.

Mediante una constante comunicación en cada paso de tiempo entre PUMITA y el modelo termohidráulico externo, los flujo y parámetros neutrónicos calculados irán evolucionando correctamente para describir el estado dinámico del reactor.

POST-PROCESAMIENTO DE RESULTADOS

Existen diferentes vías para analizar los resultados obtenidos con PUMITA . En este capítulo se describen estos mecanismos.

6.1 POST-PROCESAMIENTO EXTERNO

Se vio que los distintos procedimientos de la interfaz de PUMITA ofrecen toda la información requerida para describir el estado del reactor en cada instante de tiempo de la simulación. Una forma de procesar los resultados es por tanto a través del programa externo que emplea PUMITA , haciendo llamadas a los procedimientos de lectura de parámetros descritos en el capítulo 4 y volcando los correspondientes valores en un formato de salida adecuado.

6.2 ARCHIVO SALIDA.TXT

6.2.1 Encabezado

Para cada instante de tiempo de la simulación (más precisamente para cada llamada al procedimiento UnPaso, sea o no un cálculo estático) se muestra un encabezado con información general del cálculo, e independientemente de las opciones de impresión indicadas en ENTRADA.TXT.

La información mostrada es el instantes de tiempo, la potencia total, las componentes neutrónica y residual de la potencia, el factor de forma en el núcleo, la potencia espacial máxima junto con la ubicación donde sea alcanza la misma según canal y trozo, el factor de multiplicación y la reactividad.

6.2.2 Opciones de impresión

Si en el archivo ENTRADA.TXT se especifica el comando IMPRIMIR seguido en la misma línea por un conjunto de una o más opciones se puede modificar la información mostrada en el archivo SALIDA.TXT.

Las opciones posibles y los resultados correspondientes se listan a continuación:

- ITERACIONES: Se muestra para cada instante de tiempo calculado (más precisamente, para cada llamada al procedimiento UnPaso) información acerca del proceso iterativo para el cálculo del flujo neutrónico.

Esta información aparece organizada en 6 columnas:

- Número de iteración.
- Conv Flujo: En esta columna se muestra el parámetros EPS que determina la precisión en el cálculo del flujo neutrónico para cada iteración. Este parámetros es igual a la diferencia fraccionaria entre el flujo calculado en una iteración y el flujo en la iteración anterior. Cuando este parámetro decrece por debajo del valor de la variable PRECISION (definida por el usuario) el ciclo iterativo de cálculo es detenido.
- Con Keff: Valor de precisión relativa alcanzada por el factor de multiplicación en cada iteración. Este parámetro brinda información acerca de la precisión numérica alcanzada por Keff para el final de la simulación.
- Sigma
- Omega: Estos dos últimos parámetros forman parte del ciclo iterativo basado en el método de Sobre Relajación, mediante el cual se acelera la convergencia del flujo neutrónico a su valor final.

Además de la información por columnas, también aparecen impresos para cada instante de tiempo el método de cálculo empleado (DIRECTO o ADIABATICO), el tiempo para iteraciones y el número total de iteraciones.

- **NÚMERO:** Si la opción NUMERO está explicitada pero no la opción ITERACIONES, sólo se mostrará la información del número total de iteraciones y el tiempo para las mismas para cada instante de tiempo, omitiendo las columnas que describen el proceso iterativo explicadas en el ítem anterior.
- **DISTRIBUCION:** Se muestra la distribución de potencias específicas volumétricas por canal y trozo en MW/cm³, para cada instante de tiempo. También se muestra la potencia total (MW) del canal correspondiente, justo después del encabezado Çanal xx".
- **POR CANAL:** En este caso se muestran las potencias por canal (en MW) en un formato más compacto, y omitiendo el detalle de la distribución por trozos.

6.3 pumita2vtk

Una de las últimas modificaciones hechas a Pumita incluyen la opción de escribir la salida en formato PUMA.

La utilidad de esta opción es que el archivo SALIDA.TXT puede ser usado por el utilitario puma2vtk para obtener archivos de extensión

vtk que son graficados por el programa ParaView. Esta es una opción muy cómoda de postprocesamiento, que permite graficar distintos parámetros obtenidos con PUMITA para distintos instantes de tiempo y para distintas posiciones dentro del núcleo.

6.3.1 Formato PUMA

Para obtener los resultados en formato PUMA en el archivo SALIDA.TXT debe de especificarse la opción FORMATO PUMA en el archivo ENTRADA.TXT.

Luego de la directiva, deberá de especificarse un número entero positivo, que indica la frecuencia de impresión en Formato PUMA.

Debido a que la salida en FORMATO PUMA es bastante extensa, se da la opción de que se emplee este formato sólo cada cierta cantidad de instantes de tiempo de la simulación. Esta cantidad es lo que llamamos frecuencia de impresión.

Una vez especificados la directiva FORMATO PUMA y la correspondiente frecuencia de impresión, el archivo de SALIDA.TXT contendrá la información de ciertos parámetros de interés en formato PUMA cada una cantidad de instantes de tiempos igual a la frecuencia de impresión.

En la versión actual de PUMITA, los parámetros que se imprimen con la directiva FORMATO PUMA son:

- Potencia total (MW).
- Potencia específica volumétrica ($\frac{MW}{cm^3s}$).
- Temperatura Refrigerante(°C).
- Temperatura Combustible(°C).
- Densidad Refrigerante ($\frac{g}{cm^3s}$).
- Quemado (xxxx)
- Concentración de Xenon ($\frac{1}{cm^3s}$)
- Flujo neutrónico - Grupo1 ($\frac{n}{cm^3s}$).
- Flujo neutrónico - Grupo2 ($\frac{n}{cm^3s}$).
- Flujo neutrónico - Grupo3 ($\frac{n}{cm^3s}$).
- Flujo neutrónico - Grupo4 ($\frac{n}{cm^3s}$).
- Flujo neutrónico - Grupo5 ($\frac{n}{cm^3s}$).

Todos estos parámetros son mostrados en formato PUMA por canal y trozos, usando la numeración de simetría 1 mostrada en 3.1 para designar los canales y empleando impresiones de 18 trozos para obtener compatibilidad con modelos característicos de PUMA.

6.3.2 Empleando el utilitario *puma2vtk*

Una vez que se obtuvo el archivo `SALIDA.TXT` con formato PUMA, se debe llamar al utilitario `puma2vtk` para obtener los archivos de entrada para ParaView.

La llamada a `puma2vtk` se realiza con la siguiente instrucción a consola:

```
python puma2vtk.py GeoCarem.txt SALIDA_PUMITA.txt
```

A continuación se detalla esta intrucción:

- `python`: se llama a algún compilador instalado del lenguaje de programación python, ya que el utilitario `puma2vtk` es básicamente un archivo fuente escrito en python y para correr necesita de este compilador. Se espera que el ejecutable de este compilador esté ubicado en alguno de los directorios especificados en la variable de sistema `PATH` de la consola.
- `puma2vtk.py`: este es el archivo fuente principal del utilitario `puma2vtk`. El utilitario comprende también los archivos `func_puma2vtk` y `const.py` que deberán estar en el mismo directorio que el primero. Igualmente, este directorio deberá de estar especificado en `PATH`.
- `GeoCarem.txt`: este archivo contiene la información de geometría del reactor CAREM25. Puede ser obtenido con una corrida de puma de un modelo de CAREM25 donde se especifiquen las opciones de impresión necesarias para graficar los pasos de la red de cálculo, el mapa de regiones sobre las celdillas, las coberturas en X-FI y las coberturas axiales.
- `SALIDA.TXT`: este es el archivo de salida de PUMITA obtenido con la opción `FORMATO PUMA` pertinente.

Una vez que se corre el comando anterior con los parámetros adecuados, se generan los archivos de ParaView que permiten analizar los parámetros especificados más arriba con relativa facilidad. Se puede analizar su distribución espacial y su evolución temporal dentro de la simulación.

Parte II

MANUAL DEL PROGRAMADOR

MÓDULOS DE LA DLL

El código fuente de la DLL PUMITA se compone de tres módulos con distintas funcionalidades y que interactúan entre sí: `reactor.PAS`, `CAREMTRI.PAS` (o su versión en geometría hexagonal `CAREMHEXA.PAS`) y `interpol.PAS`

A continuación se detallan estos módulos y sus interrelaciones:

7.1 MÓDULO `reactor.pas`

La unidad `reactor.PAS` cuenta con las definiciones de los procedimientos de la *interfaz* de PUMITA . Estos procedimientos aparecen listados en el capítulo 4.

`reactor.PAS` es el primer módulo que toma el control del flujo del programa cuando la librería de Pumita es incluida en un modelo externo.

Contiene además de los encabezados de los procedimientos descritos en el capítulo 4, las instrucciones iniciales de designación de los archivos de entrada y salida; y de llamada al procedimiento inicializar contenido en `CAREMTRI.PAS` (o `CAREMHEXA.PAS`).

Ahora bien, todos los procedimientos de la interfaz de PUMITA , hacen llamadas a procedimientos más complejos, que están ocultos al usuario. Estos últimos aparecen definidos en `CAREMTRI.PAS` (o `CAREMHEXA.PAS`), que es donde ocurre la parte importante del cálculo en PUMITA .

7.2 MÓDULO `caremtri.pas` (Y SU VERSIÓN DE GEOMETRÍA HEXAGONAL `caremhexa.pas`)

La mayor parte de la acción ocurre aquí. El próximo capítulo está dedicado a describir los muchos procedimientos que están contenidos en este módulo así como las variables que definen la información calculada.

7.3 MÓDULO `interpol.pas`

Este módulo contiene procedimientos necesarios para realizar interpolación múltiple de secciones eficaces en caso de que esta sea pedida por el usuario.

INICIALIZACIÓN, VARIABLES GLOBALES

Cuando la librería de PUMITA es incluida en un programa, el procedimiento `inicializar` es corrido automáticamente. Este procedimiento está definido en el módulo `CAREMTRI.PAS` (`CAREMHEXA.PAS`)

Este procedimiento básicamente establece los valores por defecto de todas las principales variables que harán posible el cálculo neutrónico con PUMITA. Estas variables tienen carácter global y por tanto los valores almacenados en ellas perduran independientemente de que se esté ejecutando uno u otro procedimiento dentro de PUMITA.

Debido a las muchas variables globales que forman parte del código, es conveniente remitirse al código fuente y encontrar sus definiciones, acompañadas de comentarios explicando sus usos.

Dentro del mismo procedimiento `inicializar` se hace una llamada al procedimiento `ReadData` que lee información del archivo `ENTRADA.TXT`, con lo cual modifica algunas de las variables globales, pudiendo cambiar también sus valores por defecto. Para conocer las opciones que se pueden pasar a PUMITA a través del archivo de `ENTRADA.TXT` remítase a xxxx.

CÁLCULO ESTÁTICO

El grueso de las acciones de PUMITA ocurren en la unidad CAREMTRI.PAS (ó CAREMHEXA.PAS).

En este capítulo se hace una breve descripción del contenido de esta unidad a fin de facilitar futuras modificaciones.

9.1 PROCEDIMIENTO UNPASO.

Como se ha descrito anteriormente, el procedimiento UnPaso permite realizar los cálculos necesarios para la evolución de la simulación del sistema neutrónico del núcleo del CAREM.

Este procedimiento está definido en la unidad reactor.PAS, toma como argumentos el instante de tiempo actual de la simulación T y el tamaño de paso de tiempo dT.

Si $dT=0$, se estará indicando a PUMITA que se desea realizar un cálculo estático a máxima potencia y se invocará al procedimiento StaticCalculation pasándole como argumento la potencia máxima del reactor (100 MW).

Si $dT \neq 0$, se indica que se quiere realizar un cálculo dinámico y se llamará a uno de los procedimientos AdiabaticMethod o DirectMethod según el método de cálculo que se haya especificado en ENTRADA.TXT.

9.2 PROCEDIMIENTO STATICCALCULATION

En la figura 9.1 se muestra el diagrama de flujo que sigue el programa una vez que el procedimiento StaticCalculation es llamado. En este diagrama aparecen en tres columnas las variables empleadas en cada paso, los procedimientos principales, y las variables afectadas (modificadas) por estos procedimientos.

Se llama al procedimiento StaticCalculation pasándole como argumento la potencia en MW con la que se desea determinar el flujo neutrónico en cálculo estático. Dentro del procedimiento:

- Se llama al procedimiento [InterpolateXS]. En este se determinan los valores de secciones eficaces y del coeficiente de difusión necesarios para cada canal y trozo: A partir de la inserción de las barras de control (almacenadas en Ins), las tablas de secciones eficaces (almacenadas en AllTables[NMat]), el quemado de cada trozo de combustible (Burnup) y el estado neutrónico-termohidráulico del reactor (variables Tcomb,DensRerf,ConcXenon,etc.) se calculan los valores pertinentes de secciones eficaces macro-

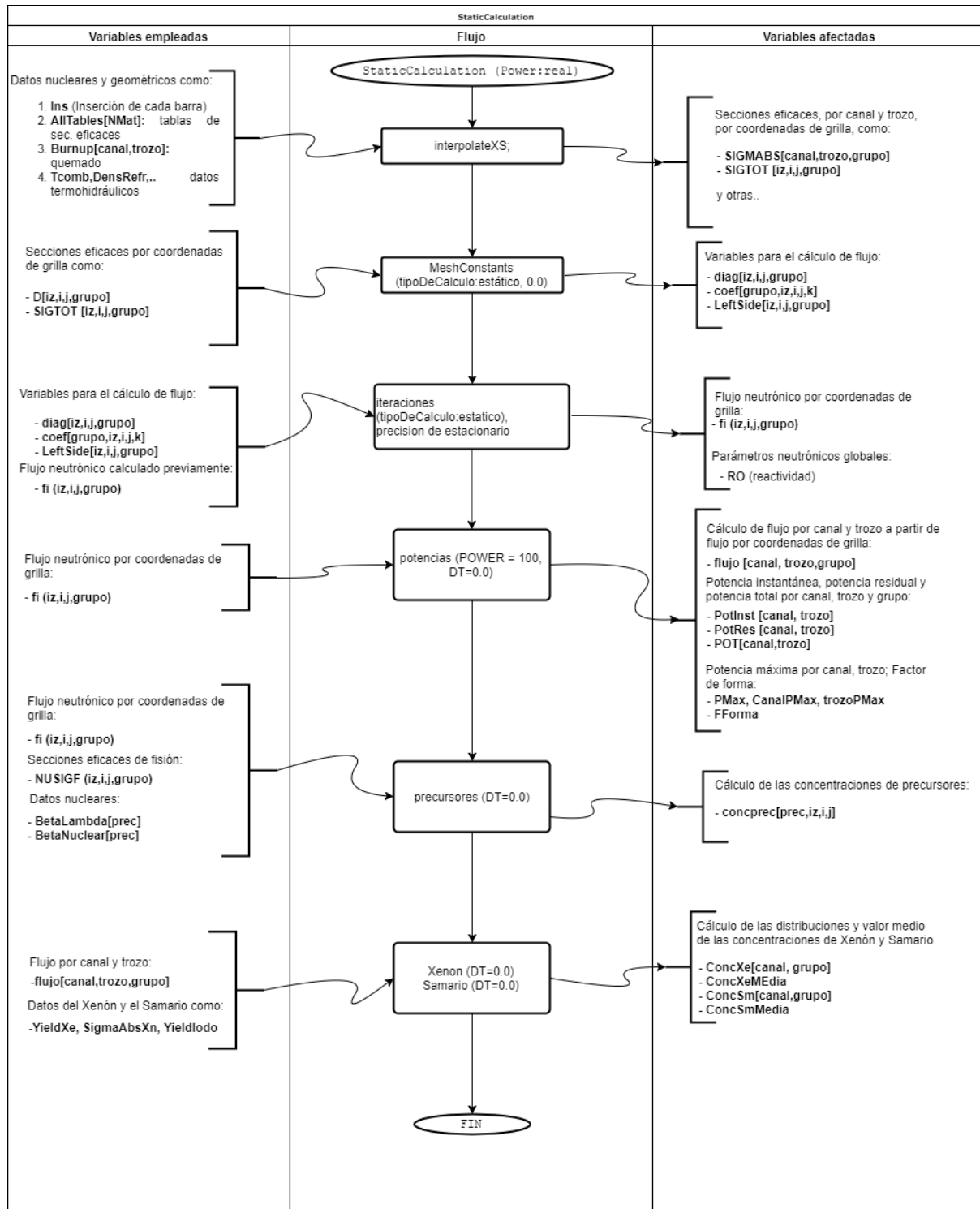


Figura 9.1: Diagrama de Flujo para el procedimiento StaticCalculation

sópicas y coeficientes de difusión por canal, trozo y grupo (variables *SIGMABS*, *SIGTOT*, etc.)

- El procedimiento *MeshConstants* calcula a partir de las secciones eficaces determinadas con *InterpolateXS* los valores de los

coeficientes que intervienen en el cálculo neutrónico con la ecuación de Difusión (variables `diag[iz,i,j,grupo]`, `coeff[grupo,iz,i,j,k]`). Estas constantes tienen una dependencia espacial con las coordenadas de la grilla (triangular o hexagonal, según sea la versión de PUMITA) y no son por canales, trozos.

- El procedimiento iteraciones, a partir de las constantes determinadas con `MeshConstants` se determina el flujo neutrónico por posiciones en grilla (`fi[iz,i,j,grupo]`) y parámetros globales como la reactividad (`R0`). Para una descripción de las ecuaciones neutrónicas y su discretización espacial (Véase [1]).
- Con el flujo neutrónico obtenido por posiciones de grilla se determinan parámetros relativos a la potencia con el procedimiento potencias: potencia instantánea `PotInst[canal,trozo]`, `PotRes[canal,trozo]`, potencias totales `POT`, factor de forma `FForma`.
- Se llama al procedimiento precursores, que a partir del flujo neutrónico calculado y otras constantes neutrónicas de cinética puntual determina la concentración de precursores según las posiciones de grilla `concprec[prec,iz,i,j]`.
- Si las correspondientes variables switch están activas, se calculan las concentraciones de Xenón y Samario con los procedimientos `Xenon` y `Samario` respectivamente. Estos necesitan de los valores de flujo neutrónico por canal y trozo (`flujo[canal,trozo,grupo]`) y datos nucleares de los isótopos en cuestión (`YieldXe`, `SigmaAbsXn`, etc.) para calcular los valores de concentración de Xenón y Samario medios y por canal y trozo (`ConcXe`, `ConcSm`).
- Se finaliza el procedimiento `StaticCalculation`, imprimiéndose los resultados pertinentes en `SALIDA.TXT` según las opciones de impresión especificadas en `ENTRADA.TXT`.

9.2.0.1 Procedimiento *InterpolateXS*

El procedimiento `InterpolateXS` permite determinar los valores correctos de las secciones eficaces macroscópicas y coeficientes de difusión que permitirán realizar el cálculo neutrónico con la ecuación de difusión. En la figura 9.1 se muestra un diagrama de flujo que nos guía a través de las etapas por las que pasa el programa una vez que entra en este procedimiento.

El ciclo mostrado en la figura 9.2 puede ser descrito en los siguientes pasos:

- Se llama al procedimiento `materiales`. En este se calculan las matrices `NumMaterialBarra`, `FraccionBarra` y `NumMaterial` para cada canal y trozo a partir de los valores de inserción de cada

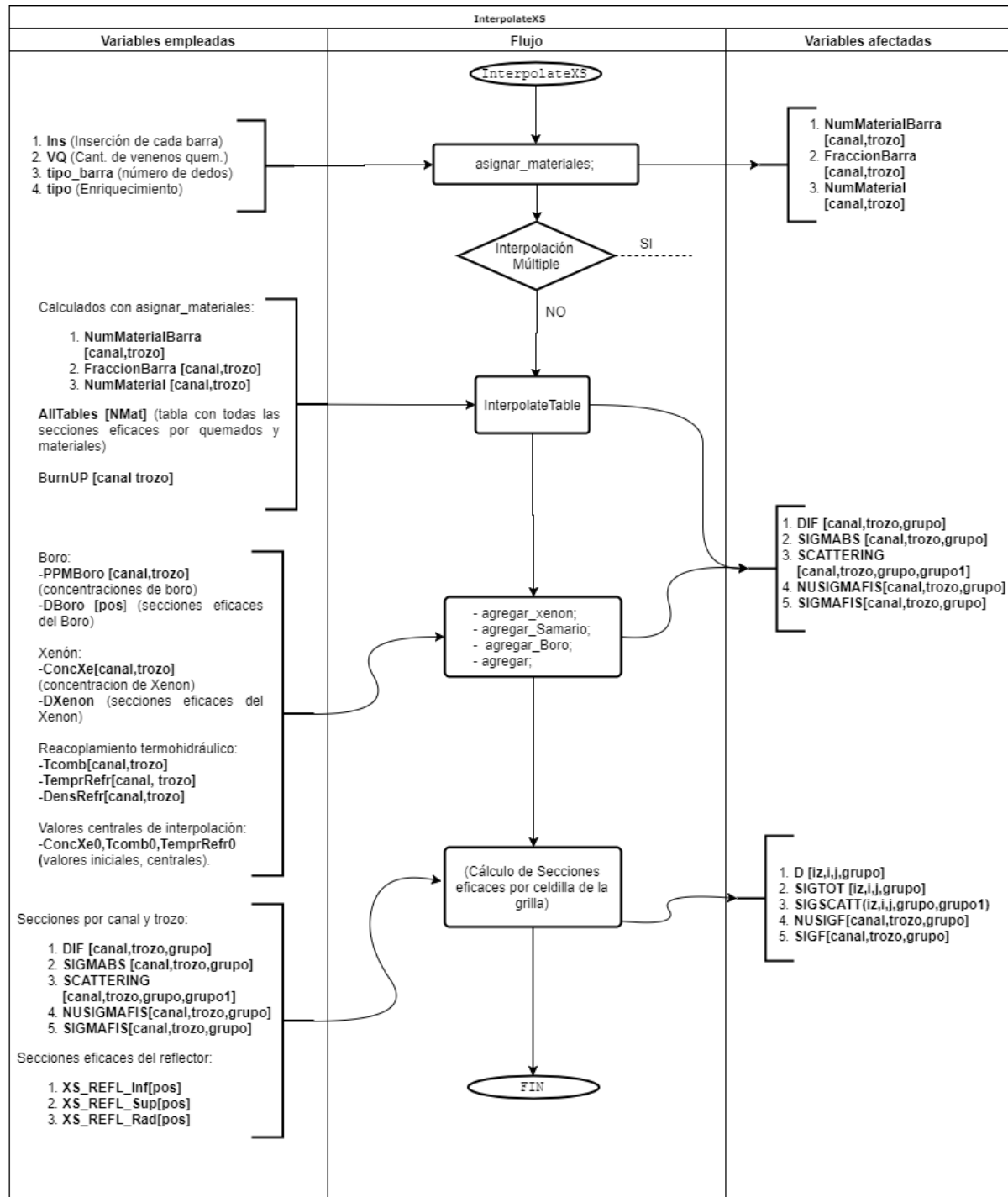


Figura 9.2: Diagrama de flujo para el procedimiento InterpolateXS

barra (*Ins*), la cantidad de venenos quemables (*VQ*), el tipo de barra (según el número de dedos), y según el enriquecimiento del combustible. Estos datos se le pasan a PUMITA con una llamada a *materiales* por cada uno de los canales del modelo de núcleo, con los datos pertinentes a cada canal. Las matrices *NumMaterialBarra*, *FraccionBarra* y *NumMaterial* contienen

información de los materiales de la barra de control insertada en un determinado trozo, la fracción de inserción (pudiendo ser de 0 a 1) y el número de material de la parte que no es barra respectivamente.

- Con la información contenida en NumMaterialBarra, FraccionBarra y NumMaterial se pasa a determinar los valores de secciones eficaces por canal, trozo y grupo (matrices DIF, SIGMABS, SCATTERING, NUSIGMAF, SIGMAFIS). Para esto se tienen cargadas las tablas de secciones eficaces por valores predefinidos de quemado en la tabla AllTables[NMAT]. El cálculo se hace interpolando los valores de estas tablas según el quemado de cada trozo (para el caso de interpolación lineal). El caso de interpolación múltiple (si está habilitado en el archivo de ENTRADA.TXT) es un tema aparte que será discutido en otro documento.
- Las secciones eficaces por canal, trozo y grupo son modificadas adecuadamente según las concentraciones de xenón, samario, boro y variables termohidráulicas si las correspondientes variables switch están activas. Los valores necesarios para estas modificaciones se encuentran en matrices por canal y trozo como PPMBoro y ConcXe.
- Finalmente, una vez que las secciones eficaces por canal y trozo tienen los valores correctos, se pasa a realizar el cálculo de las secciones eficaces por celdillas (coordenadas triangulares o hexagonales según sea el caso). Estos valores son almacenados en matrices como D[iz,i,j,grupo], SIGTOT[iz,i,j,grupo], SIGSCATT[iz,i,j,igrupo], NUS.

9.2.0.2 Procedimiento Iteraciones

Luego del procedimiento InterpolateXS, se llama al procedimiento iteraciones, que efectivamente calcula el flujo neutrónico y otros parámetros que caracterizan al sistema nuclear. La figura 9.3 muestra el correspondiente ciclo de cálculo.

El ciclo iterativo descrito en la figura 9.3 se resume en los siguientes pasos:

1. Inicialmente una etapa de inicialización, donde se actualiza el valor del flujo neutrónico calculado en el paso anterior en FI[iz,i,j,grupo] a la variable FIA[iz,i,j,grupo]. También se calcula el coeficiente de multiplicación efectivo (KE) como el cociente entre la suma de todo el flujo en el paso anterior y la suma en el paso actual.
2. Se revisa si se alcanzó el máximo número de iteraciones (MaxIter) o la precisión necesaria en el flujo neutrónico (EPS). En caso positivo se termina el ciclo, de lo contrario, se pasa a realizar una nueva iteración.

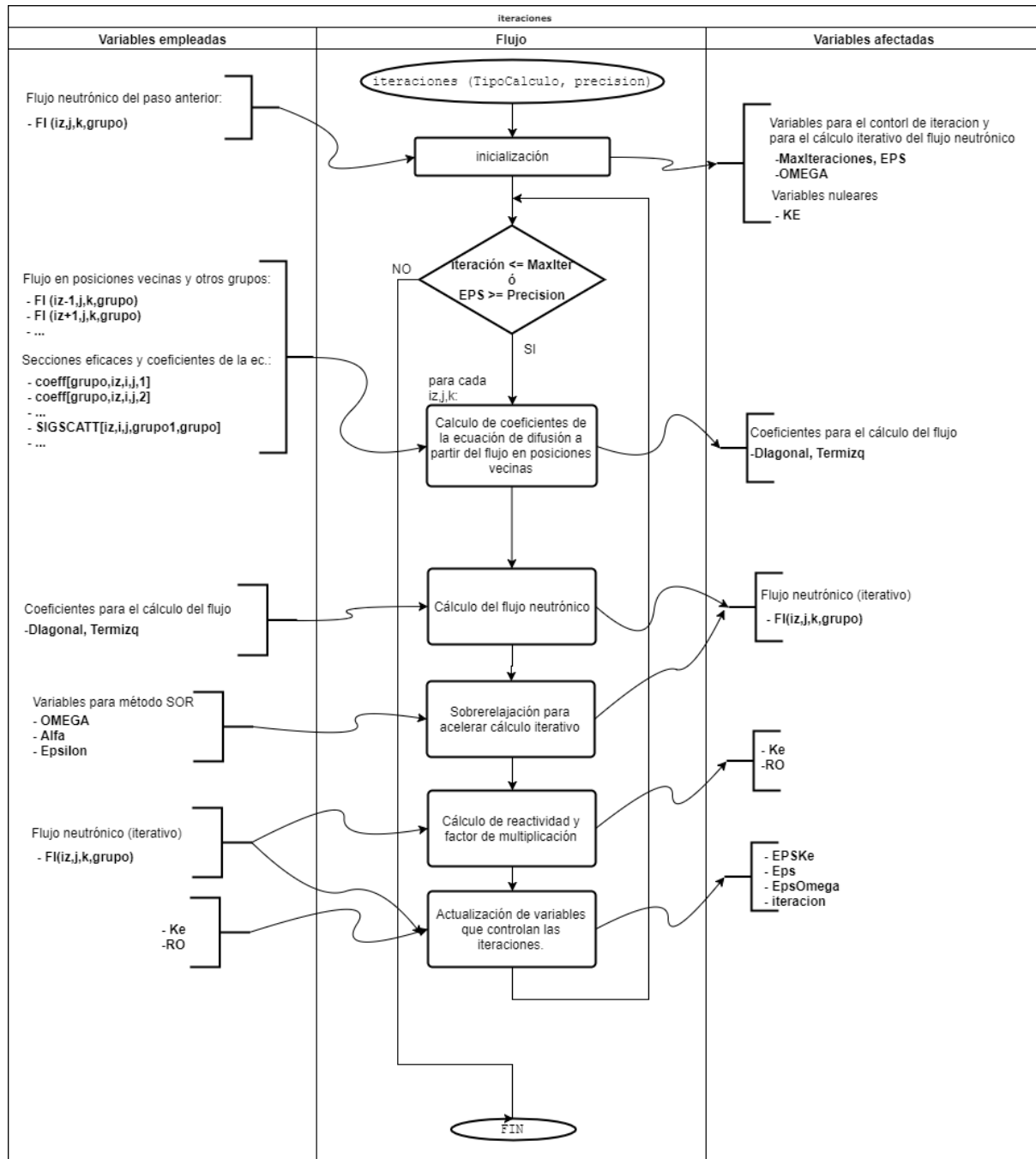


Figura 9.3: Diagrama de flujo para el procedimiento Iteraciones

- Para cada coordenada de la grilla (triangular o hexagonal) se calculan los coeficientes de la ecuación de difusión (Diagonal, Termizq, entre otros) a partir de flujos neutrónicos del paso anterior en posiciones vecinas y de los coeficientes y secciones eficaces calculados con MeshConstants e InterpolateXS respectivamente.
- Con los coeficientes determinados en el punto anterior, se resuelve el problema de autovalores para la ecuación de difusión,

del cual salen los valores de flujo neutrónico por posición de grilla.

5. A partir de parámetros numéricos OMEGA, Alfa, Epsilon se realiza un proceso de sobrerrelajación para acelerar el cálculo iterativo, mediante el cual se modifican los valores de flujo neutrónico obtenido.
6. Se calculan la reactividad y el factor de multiplicación a partir de los valores de flujo neutrónico $FI[iz,i,j,grupo]$.
7. Con los valores de flujo, reactividad y factor de multiplicación se actualizan las variables de precisión mediante las cuales se controlan las iteraciones del ciclo ($iteracion, EpsOmega, Eps, EPSKe$).
8. Se repite el ciclo.

9.2.0.3 Procedimiento Potencias

El procedimiento potencias recibe como argumentos la potencia (Power) en MW y el paso de tiempo DT. Si este último tiene valor 0.0 se entiende que se está realizando un cálculo estático y que el valor de flujo neutrónico debe de ser reescalado en a la potencia en cuestión.

El flujo de corrida correspondiente al procedimiento potencias aparece descrito en la figura 9.4. Los pasos correspondientes se detallan a continuación:

- Se llama al procedimiento pasarFlujos que a partir de los valores de flujo por celdilla $Fi[iz,i,j,k,grupo]$ calculados con iteraciones y del valor del volumen de una celdilla $VOLCeldilla$ determina los valores de flujo neutrónico por canal y trozo $flujo[canal,trozo,grupo]$. Si el procedimiento potencias fue llamado con parámetro $DT=0.0$ se calcula un factor de normalización $FNORM$ para reescalar el flujo neutrónico a la potencia prevista, y se multiplican todos los valores de flujo por dicho factor.
- Se calculan los valores de potencia instantánea $PotInst[canal,trozo]$ a partir de valores de flujo, secciones eficaces de fisión y constante de potencia por fisión.
- Se llama al procedimiento potencias_residuales que determina los valores de la matriz $PotRes$.
- Se determinan otros parámetros como la potencia máxima por canal, la potencia total y el factor de forma, todo a partir de los valores de potencia instantánea y potencia residual determinados previamente.

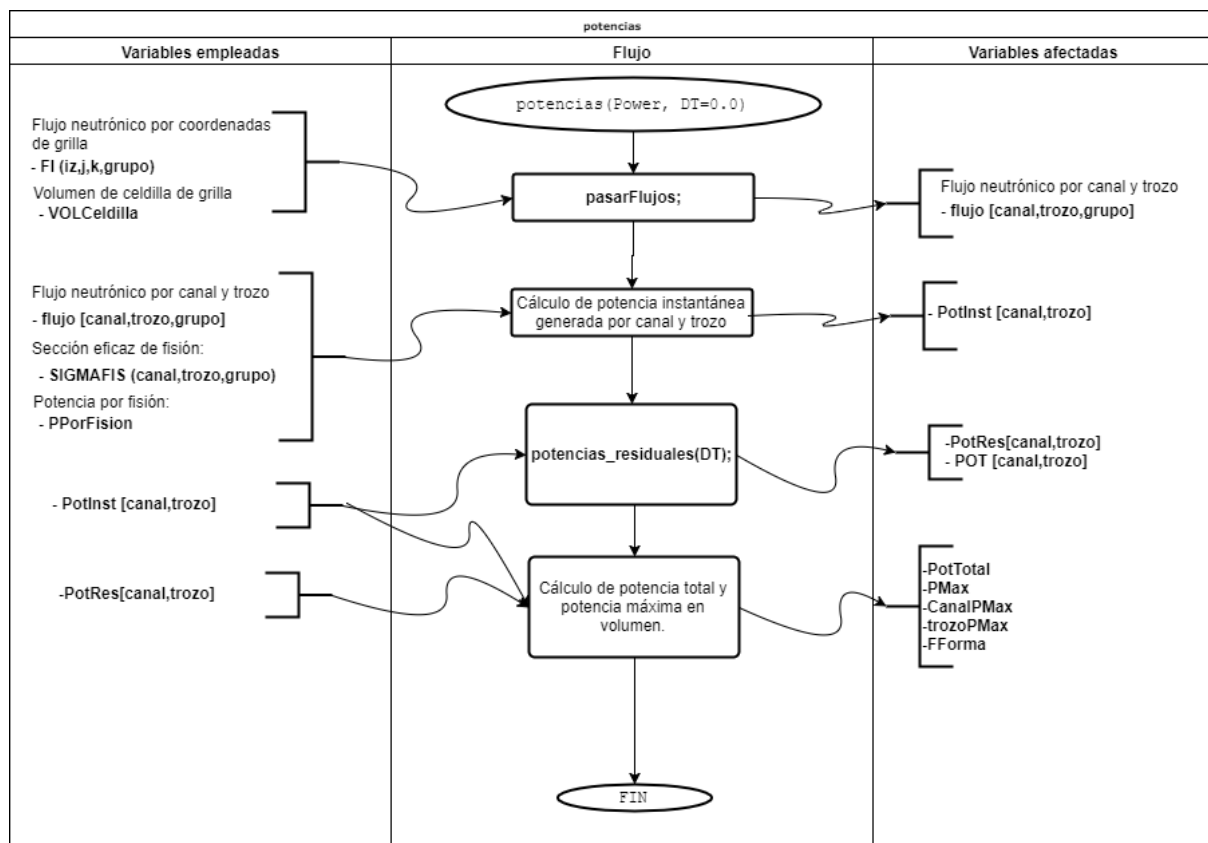


Figura 9.4: Diagrama de flujo para el procedimiento Potencias

CÁLCULO DINÁMICO

Los cálculos dinámico (del tipo adiabático o directo) emplean los mismos procedimientos que el cálculo, pero al pasar distintos argumentos (en particular el paso de tiempo no nulo), el comportamiento resultante es distinto.

En este capítulo se describen brevemente los pasos por los que pasa el programa durante los cálculos dinámicos, principalmente haciendo alusión a los procedimientos ya explicados en el capítulo anterior, y describiendo las diferencias principales.

10.1 PROCEDIMIENTO `adiabaticmethod`

El cálculo dinámico a través del método adiabático se basa en el cálculo estático y en las ecuaciones de cinética puntual. Las distintas etapas del flujo:

- Primeramente se llama al procedimiento `InterpolateXS` que cómo se vio previamente, prepara las secciones eficaces por canal y trozo para las condiciones de quemado y posiciones de barra, así como tiene en cuenta las modificaciones por parámetros neutónicos termohidráulicos si las correspondientes variables switch están activas.
- Calcula el tiempo de vida medio (`MeanLifeTime`) a partir del flujo neutrónico calculado por posiciones de grilla (`FI[iz,i,j,grupo]`), la velocidad de cada grupo de energía (`VEL[grupo]`), las secciones eficaces de fisión (`NUSIGF0[iz,i,j,grupo]`) modificadas por una división con el factor de multiplicación.
- Se llama al procedimiento `MeshConstants` esta vez con argumentos: tipo de cálculo = adiabático y el paso de tiempo `DT`. El procedimiento `MeshConstants`, para un cálculo adiabático, hará lo mismo que para el caso estático, pero al final de la corrida, modificará las secciones eficaces de fisión `NUSIGF[iz,i,j,grupo]` dividiéndolas por el factor de multiplicación del último cálculo estático realizado (`Variable KE0`). De esta forma, el método adiabático estará empleando las secciones eficaces correspondientes a un reactor crítico asociado en K .
- Se llama al procedimiento `iteraciones` con argumentos tipo de cálculo = adiabático y precisión `PrecisionCinetico` especificada en `ENTRADA.TXT`. En este caso el cálculo será similar al caso

estático, usándose solamente el tipo de cálculo para dar un formato correspondiente a la salida de impresión.

- Se llama al procedimiento `Backward_Differences` que resuelve las ecuaciones de cinética puntual (véase [1]). Además de otros parámetros como la concentración de precursores y la derivada logarítmica, este procedimiento determina el factor `cn` que permitirá modificar posteriormente el flujo neutrónico para que se avance el paso de tiempo según lo dictado por la cinética puntual.
- Se actualiza el flujo neutrónico empleando el parámetro `cn` determinado con `backward_differences`.
- Se llama al procedimiento `potencias` con argumentos `Potencia=0.0` y `Paso de tiempo=Dt`. En este caso, al pasarse el valor `0.0` de potencia, se está indicando que el cálculo que llama al procedimiento es un cálculo no estático, y se pide que *no se reescalee* el flujo a ninguna potencia en particular. Se sobreentiende que este reescalo ya se hizo durante un cálculo estático inicial. El paso de tiempo `DT` se emplea dentro del cálculo de las potencias residuales.
- Se llama al procedimiento `precursores` con argumento `Paso de tiempo = DT`. En este caso, y a diferencia del caso estático, se empleará la ecuación dinámica de precursores para calcular la nueva concentración de precursores por posición de grilla.
- Si las correspondientes variables `switch` están activas, se llaman a los procedimientos `XENON` y `SAMARIO` con argumentos `Paso de tiempo = DTXenon`. Nuevamente, el hecho de que el argumento `DT` no sea nulo, hará que se utilicen las ecuaciones dinámicas de evolución de la concentración de xenón, en vez de las estáticas.

10.2 PROCEDIMIENTO `directmethod`

El cálculo directo no hace uso de las ecuaciones de cinética puntual. En cambio, resuelve la ecuación de difusión neutrónica no estática, o sea calculando e incluyendo el término de evolución temporal a la vez que determina las secciones eficaces del reactor crítico asociado en `K`. A continuación se resumen los pasos por los que transcurre el programa durante la llamada al procedimiento `DirectMethod`.

- Llamada al procedimiento `InterpolateXS`. Véase el mismo paso en la sección 10.1.
- Llamada al procedimiento `MeshConstants` con argumentos `Tipo de cálculo = directo` y `Paso de tiempo = DT`. En este caso,

en las constantes para la ecuación de difusión se incluye la información temporal a través de la matriz `LS0` que contiene al término temporal de la ecuación. También incluye en el cálculo la información del espectro dinámico (que para los casos estático y adiabático se usaba el valor nulo) para afectar la parte `TermIzquierdo` de la ecuación. También las secciones eficaces de fisión son afectadas por división entre el factor de multiplicación del último cálculo estático (`KE`) .

- Se llama al procedimiento `CalculoCineticoEspacial`, que es la alternativa al procedimiento `Iteraciones` para los cálculos estático y adiabático. En este caso se emplea la información de los coeficientes obtenida con `MeshConstants` y el espectro dinámico calculado, para resolver iterativamente la ecuación dinámica de difusión. Como resultado se obtiene directamente el flujo neutrónico del paso siguiente a partir del flujo neutrónico calculado en el paso anterior.
- Se llama al procedimiento `potencias` con argumentos `Potencia=0.0` y `Paso de tiempo=Dt`. Véase el mismo paso en la sección [10.1](#).
- Se llama al procedimiento `precursores` con argumento `Paso de tiempo = DT`. Véase el mismo paso en la sección [10.1](#).
- Si las correspondientes variables `switch` están activas, se llaman a los procedimientos `XENON` y `SAMARIO` con argumentos `Paso de tiempo = DTXenon`. Véase el mismo paso en la sección [10.1](#).

CÁLCULO DE PARÁMETROS NEUTRÓNICO-TERMOHIDRÁULICOS

Un cálculo confiable del flujo neutrónico implica tener en cuenta parámetros neutrónicos y termohidráulicos como son las concentraciones de productos de fisión (Xenón, Samario, etc.), el estado de quemado de los materiales combustibles y las variables termohidráulicas del sistema (densidades y temperaturas en los materiales). Estos parámetros a su vez dependen del flujo neutrónico, por lo que el cálculo del mismo se basa en un proceso iterativo en el cual las interrelaciones entre los parámetros neutrónicos-termohidráulicos y el flujo den lugar a valores convergidos.

11.1 SECCIONES EFICACES

Las secciones eficaces que emplea Pumita para el cálculo del flujo neutrónico dependen de varios parámetros:

1. *Distribución de materiales:* Cada trozo de cada canal en el núcleo del CAREM tiene una composición de materiales diferente. Los distintos materiales componentes se encuentran ya previstos en el interior del código de Pumita a fin de agilizar el cálculo neutrónico. El procedimiento `asignar_materiales` permite elegir la tabla de secciones eficaces adecuada para cada canal y trozo, teniendo en cuenta la presencia o no de venenos quemables, la inserción de cada barra de control, etc.
2. *Quemado:* En el archivo de entrada `Entrada.txt` se introducen los valores de quemado por canal y trozo en el interior del núcleo del reactor. Estos valores son luego empleados para calcular las secciones eficaces correspondientes. Pumita cuenta con tablas de secciones eficaces predefinidas para un conjunto limitado de valores de quemados, por lo que el cálculo de secciones eficaces para un valor arbitrario de quemado (introducido a través de `Entrada.txt`) se realiza haciendo una interpolación lineal entre valores de secciones eficaces tabulados. Se emplea el procedimiento `InterpolateTable`.
3. *Parámetros neutrónicos termohidráulicos:* Además del quemado y la composición de materiales, Pumita brinda la posibilidad de parametrizar las secciones eficaces según los valores de ciertas magnitudes neutrónico-termohidráulicas como son la concentración de Xenón y la temperatura del combustible. Sobre cómo

se logra esto tratarán las siguientes secciones de este documento.

11.2 VARIABLES *switch*

La variable global boolean `conXenon` controla el uso o no del empleo de la concentración de xenón en el cálculo neutrónico de Pumita. De la misma manera, las variables `conSamario` y `conReacomplamientoTermohidraulico` habilitan el empleo del samario y los parámetros termohidráulicos respectivamente en dicho cálculo.

Todas estas variables son inicializadas a `false` por defecto durante la llamada del procedimiento `inicializar`. Sus valores pueden ser modificados por el usuario en el archivo de entrada `ENTRADA.txt`, que será leído por Pumita durante la llamada al procedimiento `ReadData`, dentro del seno de `inicializar`.

Para poner a `true` una o varias de las variables *switch* se deberá especificar la correspondiente opción en el archivo `ENTRADA.txt`:

```
OPCIONES QUEMADO CINETICA XENON SAMARIO REACOPAMIENTO TERMOHIDRAULICO
ADIABATICO
```

En este caso se ve, además de otras opciones, que se han especificado las palabras claves `XENON`, `SAMARIO` y `REACOPAMIENTO TERMOHIDRAULICO` que pondrán a `true` las correspondientes variables *switch*.

Las secciones eficaces microscópicas se encuentran tabuladas para un conjunto definido de valores de quemado. Para obtener los valores correspondientes a un quemado arbitrario se realiza una interpolación lineal entre valores tabulados (mediante la función `InterpolateTable`).

Las secciones eficaces tabuladas corresponden a valores predefinidos (no nulos) de concentración de xenón, concentración de samario y los parámetros termohidráulicos.

En el caso de que las variables *switch* estén a `true`, se realizarán correcciones a las secciones eficaces teniendo en cuenta la evolución de estas propiedades. Dichas correcciones se realizan de la forma:

11.3 VALORES CENTRALES

Independientemente de que se use o no la realimentación por parámetros neutrónicos-termohidráulicos (mediante las variables *switch*), las secciones eficaces tabuladas para Pumita están referidas a un conjunto de valores de estos parámetros no nulos:

Las variables son inicializadas a los valores centrales de la tabla 11.1 en el procedimiento `inicializar`. Alternativamente, en el archivo de entrada `Entrada.txt` se pueden modificar mediante el comando:

```
VALORES CENTRALES DE TABLA (ConcXE0 TComb0 DensRefr0 TempRefr0)
1.4774E+15 450.0 675.0 305.0
```

PARÁMETRO	VARIABLE EN CÓDIGO	VALOR
Temperatura de combustible	Tcomb0	450,0
Temperatura de refrigerante	TempRefr0	305,0
Densidad de refrigerante	DensRefr0	675,0
Concentración de Xenón	ConcXe0	$1,3056 * 10^{15}$
Concentración de Samario	ConcSamario0	$4,4004 * 10^{16}$

Tabla 11.1: Valores centrales de parámetros para los cuales están definidas las secciones eficaces tabuladas

Estos valores serán cargados del archivo durante la ejecución del procedimiento ReadData que ocurre dentro del mismo procedimiento Inicializar.

11.4 CÁLCULO ESTÁTICO

Para entender el acople que realiza Pumita con los parámetros neutrónico-termohidráulicos, revisaremos el caso de un cálculo estático. Para este caso, la posición de barras de control es fija en el tiempo, y por tanto se espera un flujo neutrónico estacionario. Así mismo, la potencia del reactor, la reactividad y el resto de los parámetros físicos que caracterizan el sistema deberán de ser constantes en el tiempo.

El cálculo estacionario se ejecuta en Pumita llamando al procedimiento UnPaso con argumentos $T=0.0$ y $DT=0.0$. Esto hará que el procedimiento StaticCalculation sea invocado.

El procedimiento StaticCalculation realiza el cálculo del flujo neutrónico mediante una serie de llamadas a procedimientos auxiliares:

1. procedimiento InterpolateXS: Este procedimiento calcula las secciones eficaces macroscópicas para cada canal y trozo del modelo de núcleo de CAREM. También calcula las secciones eficaces por posición de grilla (sea esta triangular o hexagonal según la versión de Pumita que se corra). Dentro del procedimiento, se accede a las secciones eficaces tabuladas y se realiza una interpolación lineal de las mismas para obtener el valor correspondiente al quemado deseado. Según las variables *switch* que estén activas en true se realiza un reajuste de las secciones eficaces acorde al correspondiente parámetro neutrónico-

termohidráulico. Por ejemplo, si la variable `conXenon` está puesta a `true`, se modifican las secciones eficaces para incorporar la diferencia debido a las distintas concentraciones de xenón en el núcleo. Así mismo, se tienen en cuenta el samario y las variables termohidráulicas si las correspondientes variables `conSamario` y `conReacoplamientoTermohidraulico` están a `true` respectivamente. Análogamente, si `conXenon` está a `false`, se dejan las secciones eficaces tal como están calculadas por la interpolación por quemado. Es importante mencionar que estas secciones eficaces tienen en cuenta una concentración de Xenón no nula, pero constante en el tiempo. Para más detalle sobre la forma en que las secciones eficaces son modificadas según cada parámetro neutrónico-termohidráulico.

2. procedimiento `MeshConstants`: En este procedimiento se modifican las secciones eficaces por posición de mallado para el caso en que se desee emplear el método de matriz de respuesta para las condiciones de contorno (a travez de la variable switch `ConMatrizDeRespuesta`). Se calculan además los coeficientes necesarios para realizar el cálculo neutrónico con la ecuación de difusión a partir de las secciones eficaces determinadas previamente. Estos coeficientes son almacenados en las matrices `coeff[grupo, iz, i, j, 1], coeff[grupo, iz, i, j, 2], ..., coeff[grupo, iz, i, j, 6], diag[iz, i, j]` para cada una de las posiciones de la grilla.
3. procedimiento `iteraciones`: Este es el procedimiento que efectivamente calcula el flujo neutrónico para cada una de las posiciones de las grillas, a partir de un procedimiento iterativo de cálculo de autovalores. El flujo neutrónico obtenido dependerá de la precisión requerida para la convergencia, así como de las secciones eficaces y coeficientes que intervienen en la ecuación de difusión, que dependen a su vez, de la distribución de materiales en el reactor y de los parámetros neutrónico-termohidráulicos. También en este procedimiento se calcula la reactividad del sistema neutrónico a partir de la evolución del flujo de una iteración a la siguiente.
4. procedimiento `potencias`: Con el flujo neutrónico calculado se determinan las potencias neutrónica, residual y total, empleando las secciones eficaces de fisión para cada posición de la grilla. También se determinan valores de interés como el factor de pico y la potencia máxima.
5. procedimiento `precursores`: En este procedimiento se calcula la concentración de precursores en cada posición de la grilla a partir del flujo neutrónico previamente obtenido.
6. procedimientos `XENON` y `SAMARIO`: Si las correspondientes variables switch están a `true` se calculan en estos procedimientos las

concentraciones de xenón y samario respectivamente para cada canal y trozo del núcleo. Estos valores se obtienen a partir del flujo neutrónico obtenido, de las secciones eficaces de absorción del Samario y Xenón (tabuladas acordemente) y de parámetros como la constante de decaimiento de cada isótopo.

Al finalizar toda la secuencia del procedimiento UnPaso se tendrán debidamente calculados las variable de interés para describir el sistema nuclear: el flujo neutrónico, la potencia, la reactividad, las secciones eficaces, las concentraciones de precursores, las concentraciones de xenón y samario, etc.

Es importante señalar que luego de una llamada a UnPaso, el resultado del flujo neutrónico puede no encontrarse completamente acoplado con los parámetros neutrónicos termohidráulicos. Esto debido a que hay una dependencia mutua entre uno y otros, a través de las secciones eficaces. Por tanto, es necesario hacer varias llamadas a UnPaso a fin de obtener un flujo debidamente *acoplado* con los parámetros neutrónicos termohidráulicos.

11.5 CÁLCULO ESTÁTICO DE XENÓN

11.5.1 Variables empleadas en el código

1. ConXenon: boolean: Variable *switch* que permite al usuario especificar si se tendrá en cuenta o no la evolución del xenón y el reacoplamiento del mismo con el flujo para el cálculo neutrónico.
2. ConcXE0: real: Contiene el valor central de concentración de xenón. Las secciones eficaces y coeficientes de difusión que emplee Pumita deben de estar calculadas teniendo en cuenta la presencia de esta concentración de xenón en los materiales.
3. ConcXE[canal,trozo]: matriz de reales: Contiene los valores de la concentración de xenón para cada trozo dentro de cada canal del núcleo del reactor.
4. flujo[canal,trozo,grupo]:matriz de reales: Contiene el valor del flujo neutrónico para cada trozo dentro de cada canal del núcleo del reactor, para cada grupo.
5. FI[iz,i,j,grupo]: matriz de reales: Contiene el valor de flujo neutrónico para cada posición (iz,i,j) de la retícula (triangular o hexagonal según el modelo de Pumita empleado) del núcleo del reactor.

11.5.2 Cálculo del Xenón

El procedimiento inicializar dentro de CAREMTRI.pas (o CAREMHEXA.pas) se encarga de inicializar todas las variables globales.

Las variables ConcXe0 y $\text{concXe}[\text{canal}, \text{trozo}]$ corresponden al valor de concentración de xenón de referencia y a la matriz de concentraciones de xenón en el reactor por canal y trozo respectivamente. La primera es inicializada a $1,3056 * 10^{15}$ dentro del procedimiento `inicializar`, aunque puede ser cambiada a través del archivo de entrada `Entrada.txt`. La segunda es inicializada para todos los canales y trozos al mismo valor $1,3056 * 10^{15}$.

El procedimiento `InterpolateXS` calcula las secciones eficaces y el coeficiente de difusión para el quemado correspondiente a cada trozo y canal del reactor. Se realiza una interpolación entre los valores tabulados en un archivo externo para quemados predefinidos. Las secciones eficaces y coeficientes de difusión obtenidos para cada grupo son *macroscópicas* y tienen en cuenta las concentraciones de los distintos materiales que componen el reactor. Además, estas secciones eficaces están definidas para un valor no nulo de concentración de Xenón. Este valor, al que llamaremos valor central, es usado como referencia para la modificación posterior de secciones eficaces como se describe en el próximo párrafo.

Si la variable `switch conXenon=true`, dentro del procedimiento `InterpolateXS` se llama al procedimiento `agregar_Xenon` pasándole como argumentos la posición `N00=241` de tabla de secciones eficaces y la diferencia entre la concentración de xenón por canal y trozo y la concentración de xenón de referencia $\Delta X = \text{concXe}[\text{canal}, \text{trozo}] - \text{concXe0}$.

Dentro del procedimiento `agregar_Xenon` todas las secciones eficaces calculadas previamente mediante interpolación por quemado, son modificadas según las siguientes ecuaciones:

$$\begin{aligned} D &= D_0 + \Delta X * \frac{\partial D}{\partial X} \\ \Sigma_{\text{abs}} &= \Sigma_{\text{abs},0} + \Delta X * \sigma_{\text{abs}}^{\text{Xe}} \\ \Sigma_{\text{scatt}} &= \Sigma_{\text{scatt},0} + \Delta X * \sigma_{\text{scatt}}^{\text{Xe}} \\ \Sigma_{\text{fis}} &= \Sigma_{\text{fis},0} + \Delta X * \sigma_{\text{fis}}^{\text{Xe}} \end{aligned}$$

donde $D_0, \Sigma_{\text{abs},0}, \Sigma_{\text{fis},0}, \Sigma_{\text{scatt},0}$ y $D, \Sigma_{\text{abs}}, \Sigma_{\text{fis}}, \Sigma_{\text{scatt}}$ son los coeficientes de difusión, las secciones eficaces macroscópicas de absorción, de scattering y de fisión antes y después de la incorporación del xenón respectivamente. Además ΔX es la diferencia entre la concentración de xenón real y la concentración de referencia para la cual están referidas las secciones eficaces previamente mencionadas. $\frac{\partial D}{\partial X}, \sigma_{\text{abs}}^{\text{Xe}}, \sigma_{\text{scatt}}^{\text{Xe}}, \sigma_{\text{fis}}^{\text{Xe}}$ son los valores microscópicos del coeficiente de

difusión y las secciones eficaces del xenón, todos tabulados en archivo externo.

Luego, dentro del procedimiento `StaticCalculation` se realiza el cálculo del flujo neutrónico con las secciones eficaces ya modificadas. Para esto se emplea el procedimiento `iteraciones`.

Si la variable `switch conXenon` está a `true`, se llama al procedimiento `Xenon` que calcula la concentración de xenón por canal y trozo a partir del flujo neutrónico previamente obtenido y las secciones eficaces microscópicas del xenón tabuladas.

11.5.3 *Cálculo estático del Samario y de los Parámetros termohidráulicos*

De manera análoga a como se describió con el Xenón, `PUMITA` es capaz de incorporar la influencia del Samario, del Boro y de los parámetros termohidráulicos a las secciones eficaces calculadas en cada paso de tiempo.

Para esto deberán activarse las correspondientes variables `switch`, tal como se describió previamente.

La forma en que las secciones eficaces son calculadas es completamente análoga al caso del xenón, y por tanto no necesitan ser incluidas aquí: una simple inspección del código fuente permitirá encontrar las equivalencias en los métodos.

Parte III

APÉNDICES

DESCRIPCIÓN DEL FORMATO DE LAS TABLAS DE SECCIONES EFICACES

Se utilizan las tablas empleadas en CITVAP para la gestión de combustible. Se asigna un número de tabla para cada una de las que están descritas en la Tabla 3.1 y cada una de éstas contendrá para cada quemado bloques de $8 \cdot ng$ valores en la siguiente forma (ng = número de grupos):

1. El valor de quemado.
2. Un bloque de 40 valores con las secciones eficaces para la temperatura de combustible de 450°C , la temperatura de refrigerante de 305°C y la densidad de refrigerante de $675\text{Kg}/\text{m}^3$.
3. Un bloque de 40 valores con las derivadas de cada sección eficaz respecto a la temperatura del combustible.
4. Un bloque de 40 valores con las derivadas de cada sección eficaz respecto a la temperatura del refrigerante.
5. Un bloque de 40 valores con las diferencias entre los valores de las secciones eficaces a $400\text{Kg}/\text{m}^3$ y los valores de las secciones eficaces del bloque 2.
6. Un bloque de 40 valores con las diferencias entre los valores de las secciones eficaces a $550\text{Kg}/\text{m}^3$ y los valores de las secciones eficaces del bloque 2.
7. Un bloque de 40 valores con las diferencias entre los valores de las secciones eficaces a $753\text{Kg}/\text{m}^3$ y los valores de las secciones eficaces del bloque 2.
8. Un bloque de 40 valores con las derivadas de cada sección eficaz respecto a la concentración de xenón.
9. Un bloque de 40 valores con las derivadas de cada sección eficaz respecto a la concentración de samario.

El orden en que se dan los valores de cada bloque de 40 es el que corresponde para el formato PUMA, especial para teoría de difusión:

$$\begin{aligned} & (D(g), \quad g=1, \dots, NG), \\ & ((SIG(g, g1), \quad g1=1..NG), \quad g=1, \dots, NG), \\ & (NUSIGF(g), \quad g=1, \dots, NG), \quad (SIGF(g), \quad g=1, \dots, NG) \end{aligned}$$

es decir, primero van los coeficientes de difusión, luego siguen las matrices SIG y finalmente los valores de NU multiplicados por la sección eficaz de fisión, finalizando con las secciones eficaces de fisión.

en donde:

$$\text{SIG}(g, g1) = \text{SIGA}(g) \text{ para } g = g1$$

$$\text{SIG}(g, g1) = \text{SCAT}(g, g1) \text{ para } g \neq g1$$

En los bloques 3 a 9 en los primeros 5 valores las derivadas o diferencias se dan respecto a la sección eficaz de transporte y no al coeficiente de difusión.

Las concentraciones de Xenón y Samario se miden en neutrones/cm³ en la celda.

DESCRIPCIÓN DEL ARCHIVO DEL ESTADO INICIAL DEL REACTOR

El archivo ENTRADA.TXT describe el estado inicial del reactor al comenzar a hacer la simulación.

```
*****
***** Estado inicial para la simulacion *****
*****
```

```
OPCIONES QUEMADO CINETICA XENON SAMARIO REACOPPLAMIENTO
      TERMOHIDRAULICO ADIABATICO
```

```
PASO DE TIEMPO
```

```
0.2
```

```
ITERACIONES MAXIMO
```

```
30
```

```
TABLAS
```

```
TCOMUN\_AMPLIADA.TXT
```

```
INTERPOLACION LINEAL
```

```
PRECISION PARA EL CALCULO ESTATICO
```

```
1.0E-8
```

```
PRECISION PARA EL CALCULO CINETICO
```

```
1.0E-5
```

```
Potencia de Fision
```

```
192.0
```

```
COEFICIENTE DE SOBRERRELAJACION
```

```
1.6
```

```
INTERCALADOS
```

```
QUEMADOS
```

```
CAN E1
```

```
13343.6  16013.3  18104.6  19115.7  19213.5  18321.6
```

```
17711.6  16362.5
```

```
14657.0  12811.1  10664.3  9056.5  8125.5  6150.7
```

```
CAN D2
```

```
14329.4  17036.0  18624.3  19344.0  19192.3  18083.3
```

```
17144.9  15222.2
```

```
13446.2  11611.3  9633.6  8224.7  7447.0  5774.1
```

```
... siguen de este modo los quemados...
```

TEMPERATURA DEL COMBUSTIBLE

CAN E1

367.08	374.92	376.79	376.26	373.10	367.94
363.57	359.77				

355.69	352.05	348.73	342.88	318.11	318.09
--------	--------	--------	--------	--------	--------

CAN D2

391.80	398.66	398.01	393.05	386.93	372.46
366.11	361.36				

356.53	352.56	349.49	343.53	319.79	319.76
--------	--------	--------	--------	--------	--------

CAN F1

388.34	395.12	394.91	391.08	387.07	380.74
374.98	369.69				

364.03	359.50	356.28	348.80	321.03	321.04
--------	--------	--------	--------	--------	--------

.. siguen de este modo las temperaturas del combustible...

TEMPERATURA DEL REFRIGERANTE

CAN E1

289.26	292.05	295.05	298.13	301.09	303.95
306.60	309.07				

311.33	313.34	315.15	316.77	318.07	318.09
--------	--------	--------	--------	--------	--------

CAN D2

290.66	294.21	297.88	301.50	304.78	307.77
310.24	312.41				

314.33	315.98	317.43	318.72	319.75	319.75
--------	--------	--------	--------	--------	--------

.. siguen de este modo las temperaturas del refrigerante...

DENSIDAD DEL REFRIGERANTE

CAN E1

745.82	739.57	732.66	725.37	718.19	711.09
704.40	698.06				

692.22	686.96	682.18	677.78	673.90	674.18
--------	--------	--------	--------	--------	--------

CAN D2

742.67	734.56	725.87	716.54	707.58	699.00
693.87	688.61				

683.63	679.24	675.19	671.26	668.45	668.98
--------	--------	--------	--------	--------	--------

.. siguen de este modo las densidades del refrigerante...

BIBLIOGRAFÍA

- [1] Grant C. *Descripción de los métodos y modelos utilizados en el sistema PUMA*. - ITE-06 REC-268.
- [2] Etchepareborda A. Lopasso E. *ACUERDO DE TRABAJO: Desarrollo del código de cálculo PUMITA para CAREM - ATR-CAREM₂₅N-1-B1010*.
- [3] Weir A. Lucero Lorca J.P. *Informe preliminar de seguridad del núcleo del reactor CAREM 25 - IS-CAREM₂₅N-1-B1010*.
- [4] Torres L. Serra O. *Actualización a junio de 2010 de los cálculos de núcleo del reactor CAREM 25 - IN-CAREM₂₅N-17-B1010-ro*.