



Análisis a Nivel de Varilla de Combustible de un Accidente de Inserción de Reactividad con el Código Acoplado Paralelo pCTF/PARCSv2.7

E. Ramos, A. Abarca, J. E. Roman y R. Miró
Universitat Politècnica de València
Camí de Vera s/n, 46021 València, Spain

ramos@iqn.upv.es, aabarca@iqn.upv.es, jroman@iqn.upv.es, rmiro@iqn.upv.es

Resumen – La descripción detallada del núcleo del reactor tiene cada vez más importancia para los análisis de seguridad de los Reactores de Agua Ligera (LWR). Gracias a los avances en la tecnología informática, los diferentes diseños de elementos combustibles pueden ser modelados de manera realista mediante códigos informáticos de simulación, como en el caso de los códigos neutrónicos y termohidráulicos en 3D. Este tipo de cálculos requiere de códigos termohidráulicos altamente eficientes, para poder obtener resultados en tiempos razonables utilizando modelos muy detallados.

En este trabajo se presentan los esfuerzos realizados sobre el código termohidráulico 3D de subcanal COBRA-TF (CTF) para reducir su tiempo de respuesta cuando se simulan grandes reactores nucleares con alto nivel de detalle en los modelos. Para ello se ha desarrollado una versión paralela de dicho código, llamada pCTF, utilizando el estándar MPI (Message Passing Interface). El objeto del estudio es demostrar la capacidad del código paralelo acoplado desarrollado pCTF/PARCS de simular grandes núcleos de reactor a nivel de varilla de combustible y en un tiempo de simulación aceptable.

Para demostrar la capacidad del código se ha seleccionado un transitorio tipo RIA (Reactivity Insertion Accident) que tiene lugar en un reactor tipo PWR de tres lazos. Como resultados se presentan los principales parámetros de seguridad calculados en el canal caliente por el código acoplado, obteniendo unos resultados best estimate para este tipo de transitorio.

1. INTRODUCCIÓN.

La descripción detallada de los elementos de combustible está creciendo en importancia en los análisis de seguridad nuclear de reactores de agua ligera (LWR), con objeto de contar con modelos detallados de elementos de combustible para obtener de manera precisa los parámetros de seguridad (PCT, entalpía del combustible o DNBR) a nivel de varilla. El nivel de detalle deseado para los modelos termo-hidráulicos puede alcanzarse utilizando los llamados códigos de subcanal. Para realizar estos análisis en un tiempo razonable se requiere de un código de subcanal eficiente, siendo si cabe más importante en el caso de códigos de dos fluidos como CTF [1, 2, 3], debido a la gran cantidad de complejas ecuaciones a resolver.

En este trabajo se presenta una versión paralela del código CTF (pCTF) de modo que puedan obtenerse importantes *speedups* en pequeños clústeres de computadores. En particular, se ha utilizado una paralelización de paso de mensaje (con MPI) seguida de una aproximación por descomposición en subdominios y utilizando los solucionadores lineales paralelos proporcionados por PETSc [4].

El trabajo se organiza de la siguiente manera. Primero se describe la estructura del código y la paralelización, posteriormente se presenta el acople entre pCTF y PARCSv2.7., y por último se presentan los resultados obtenidos en la simulación de un transitorio de caída de barra de control que tiene lugar en el núcleo de un reactor de agua a presión (Pressurized Water Reactor (PWR)) de tres lazos.



2. DESCRIPCIÓN DE CTF.

COBRA-TF (Coolant Boiling in Rod Arrays Code – Two Fluids), abreviado como CTF, es un código termohidráulico de subcanal (permite resolver problemas a nivel de varilla de combustible nuclear) que utiliza dos campos y tres fases para modelar el flujo bifásico (agua+vapor)]. Las tres fases son el vapor, el líquido continuo y el líquido presente a modo de gotas dentro del vapor en ciertos regímenes. Cada fase utiliza un conjunto de tres ecuaciones tridimensionales para la masa, momento y energía, con una excepción: se utiliza una ecuación de la energía para ambas fases líquidas, la de líquido y la de las gotas presentes en la fase vapor. La formulación de dos fluidos emplea un conjunto separado de ecuaciones de conservación y relaciones constitutivas para cada fase. Los efectos de una fase en otra se tienen en cuenta en los términos de interacción que aparecen en la correspondiente ecuación de gobierno. Las ecuaciones de conservación tienen la misma forma para cada fase; solamente se diferencian en las relaciones constitutivas y las propiedades físicas.

2.1. Estructura del código paralelo pCTF.

Con el objetivo de ser capaz de resolver grandes problemas en un tiempo razonable con el código de subcanal CTF se ha utilizado computación paralela. A través de la paralelización, numerosos procesadores pueden utilizarse cooperando para obtener una solución única al problema, y por consiguiente reduciendo el tiempo computacional así como la cantidad de memoria disponible en el clúster.

Básicamente, la paralelización consiste en dividir el trabajo computacional en sub-tareas separadas que se asignan a diferentes procesadores. Para conseguir una paralelización efectiva deben tenerse en cuenta dos aspectos importantes: (1) debe paralelizarse todo el esquema de resolución del código para no crear cuellos de botella, y (2) la memoria debe manejarse de manera apropiada, dividiendo las variables entre los diferentes procesos. Para ello se ha considerado una paralelización a nivel de subdominio basada en la descomposición axial del dominio del problema utilizando el paradigma del paso de mensajes, en particular con MPI [6].

2.2. Descomposición en subdominios.

La aproximación de descomposición del dominio para la paralelización es apropiada para el tipo de cálculos realizados por CTF. Se ha elegido una descomposición por subdominios en la dimensión axial, esto es, a cada proceso MPI se asigna un grupo de niveles axiales contiguos del dominio simulado. Normalmente el número de niveles axiales es bastante pequeño, de modo que la escalabilidad del código paralelo está limitada por esta dimensión. Esto podría compensarse con un segundo nivel de paralelismo basado en la computación en *multi-thread* asociados a cada subdominio.

Debido al esquema de discretización de CTF, existe una gran cantidad magnitudes dependen de los valores de las celdas vecinas. Por tanto, es necesario representar el estado de las variables de forma que cada procesador tenga suficiente espacio de memoria para asignar todas las variables de los niveles axiales asociados y el de las celdas *virtuales*. Estas celdas *virtuales* contienen memoria extra para almacenar los datos necesarios procedentes de los procesos vecinos. Siempre que sea necesario durante el cálculo, la información de estas celdas *virtuales* será actualizada con los valores de los procesos vecinos, y esto se realiza a través de comunicación punto-a-punto.

2.3. Solucionador Lineal.

Los desarrollos implementados comienzan a partir de una versión de CTF que utiliza para resolver el sistema lineal de ecuaciones asociadas con el Jacobiano de cada celda la librería SPARSKIT de métodos iterativos de Krylov. Para la versión paralela se ha sustituido SPARSKIT por PETSc, que proporciona solucionadores lineales paralelos que ajustan de manera adecuada con el paradigma de la descomposición del dominio. Además PETSc es mucho más flexible a la hora de seleccionar diferentes métodos iterativos y preconditionadores.

PETSc (Portable, Extensible Toolkit for Scientific Computation) [4, 5] es un marco para la solución paralela de ecuaciones en derivadas parciales. PETSc está disponible gratuitamente y se utiliza en multitud de áreas de aplicación incluyendo la ingeniería nuclear.

Para resolver sistemas lineales de ecuaciones, PETSc proporciona una amplia variedad de métodos iterativos, como el GMRES, que puede ser combinado con diferentes preconditionadores como el Jacobi o el Jacobi por bloques. Un estudio anterior demostró que el BiCGStab(l) con el Jacobi por bloques (con LU incompleta para cada subdominio) presenta el mejor comportamiento, de modo que se utilizará este método numérico en los cálculos de pCTF que se presentan en este trabajo.

3. CÓDIGO ACOPLADO CTF/PARCSv2.7.

El esquema de acople explícito entre los códigos CTF y PARCS se basa en un intercambio de datos externo. El algoritmo de acople utiliza las rutinas de la *General Interface* (GI) implementadas en PARCS y nuevas rutinas añadidas en CTF para posibilitar la comunicación entre los códigos utilizando la *Parallel Virtual Machine* (PVM), como puede observarse en el esquema de la Figura 1.

Originalmente, CTF poseía una rutina de *auto-mapping* para adaptar los datos a las diferentes nodalizaciones empleadas por ambos modelos termo-hidráulico y neutrónico. Sin embargo en modelos más complejos de CTF, como es desarrollado para este estudio, que incluye los plenos del núcleo, podría ser necesario el uso de archivos externos para proporcionar la correspondencia entre ambas nodalizaciones (*mapping*). La capacidad de incluir estos archivos, denominados MAPTAB, se ha añadido al acople original entre ambos códigos.

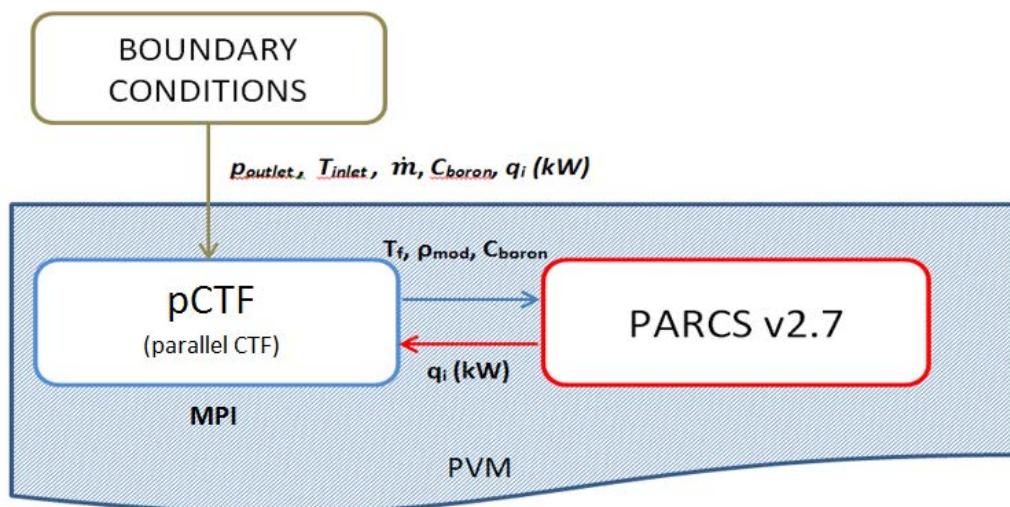


Figura 1. Esquema de acople y flujo de datos.



4. TRANSITORIO DE CAÍDA DE BARRA.

El núcleo modelado corresponde con el de un reactor de agua a presión de tres lazos. El núcleo tiene 177 elementos combustibles, siendo el número total de varillas por combustible de 236, además de 20 tubos guía para las barras de control. Las condiciones operacionales corresponden con el HFP (*Hot Full Power*), donde la potencia nominal y el caudal másico a través del núcleo son de 3010 MWth y 15605.6 kg/s respectivamente en el principio del ciclo (Beginning of Cycle (BOC)).

La inserción de la barra de control se define en el archivo de entrada de PARCS. La caída dura 2.1 segundos, y comienza a los 5.0 segundos de simulación en un transitorio de una duración total de 30.0 segundos. Los primeros 5.0 segundos son de transitorio nulo, con el propósito de asegurar unas condiciones estacionarias estables antes al comienzo del transitorio. La Figura 2 muestra la evolución de la inserción de la barra de control durante el transitorio, siendo 340cm la posición completamente extraída y 0cm completamente insertada de la barra de control.

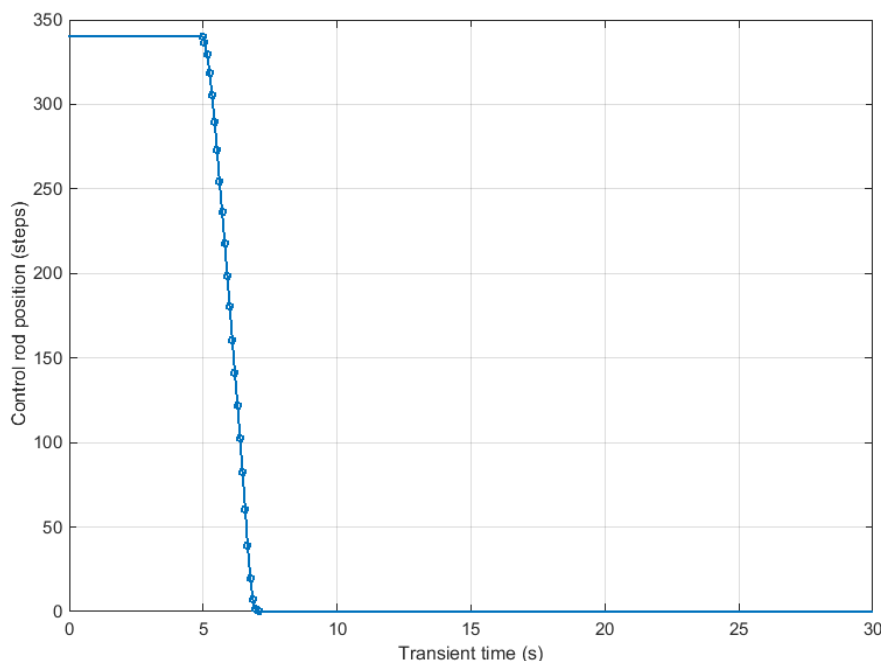


Figura 2. Evolución temporal de la posición de la barra de control (en cm).

5. MODELOS Y METODOLOGÍA.

5.1. Modelo termo-hidráulico.

El modelo varilla a varilla de CTF contiene 45313 canales termo-hidráulicos, 45312 representan el núcleo activo (coloreado en azul en la Figura 3) y 1 para el conjunto del bypass de núcleo (coloreado en rojo en la Figura 3). Axialmente el dominio se divide en 34 nodos igualmente espaciados. Las 41772 varillas de combustible que comprenden el núcleo del reactor están representadas por igual número de estructura térmicas, así como los 3540 tubos guía representados por sus respectivas estructuras no calefactadas. En resumen, el núcleo activo está modelado por una malla tridimensional de 45312 x 34 nodos con las estructuras térmicas de las varillas de combustible y tubos guía conectadas a sus respectivos canales hidráulicos. La Figura 3 muestra una representación esquemática del modelo de núcleo de CTF.

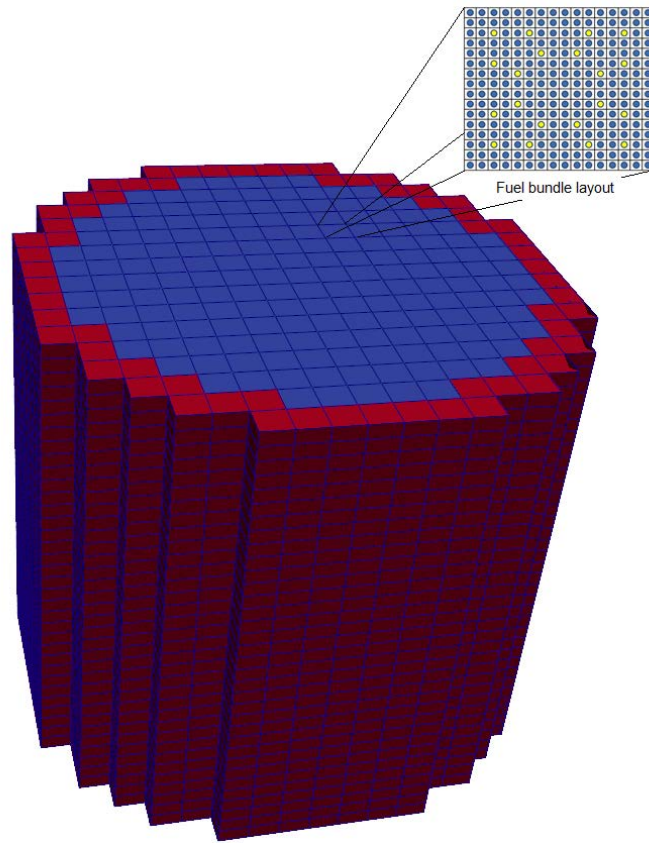


Figura 3. Malla 3D del modelo termo-hidráulico de CTF.

5.2. Modelo neutrónico.

El modelo neutrónico se ha desarrollado utilizando el ampliamente probado código de difusión neutrónica PARCSv2.7, que se acopló con CTF en trabajos anteriores. Como puede observarse en la Figura 4, el modelo consiste en un mallado tridimensional de 241 x 34 nodos, en los que el núcleo activo corresponde con la malla centra de 177 x 32 nodos. Para las tablas de secciones eficaces se han considerado 1379 composiciones isotópicas diferentes, según el diseño del combustible, el quemado y la potencia relativa de cada nodo, 1376 correspondientes con el núcleo activo y 3 con los reflectores superior, inferior y circundante respectivamente. Las secciones eficaces empleada en el cálculo neutrónico, proporcionadas por Iberdrola Generación Nuclear S.A. y CNAT, han sido generadas utilizando la metodología SIMTAB desarrollada en la *Universitat Politècnica de València* (UPV) junto con Iberdrola S.A.U. Esta metodología se alimenta de datos de los códigos CASMO4-SIMULATE y genera las tablas de secciones eficaces en función de la temperatura del combustible, densidad del moderador, concentración de boro y fracción de control.

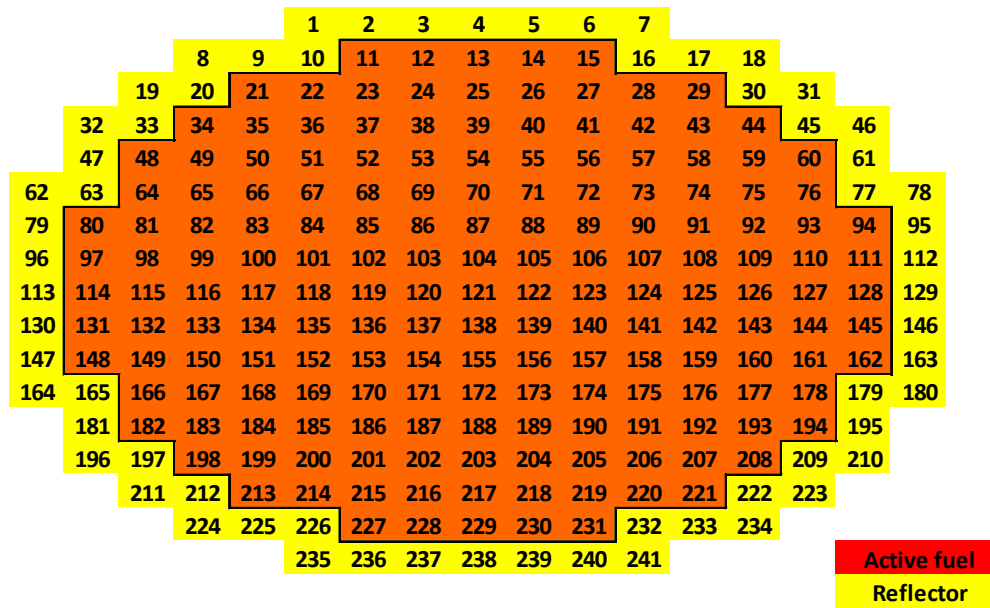


Figura 4. Esquema radial del modelo neutrónico.

5.3. Power Density Detectors.

Con el objetivo de obtener la respuesta de las *Power Density Detectors* (PDDs) utilizando el flujo neutrónico calculado por PARCS, se utiliza la Ec. 1, donde F_{ID} es un factor de conversión para cada detector, que corresponde con el valor medio de potencia dado por el detector en el caso estacionario, y Φ es el flujo térmico calculado en la posición del detector.

$$P_{ID} = \frac{\phi}{\phi_{t=0}} \times F_{ID} \quad (1)$$

La posición real de los PDDs en el interior del núcleo se muestra en la Figura 5, donde también puede apreciarse, resaltada en rojo, la posición ocupada por la barra de control insertada. La posición radial de los PDDs corresponde con un único nodo neutrónico, sin embargo axialmente pueden pertenecer a dos nodos consecutivos, y por ello se multiplicará el flujo por el apropiado factor de pesado para obtener la respuesta del detector. Estos factores de pesado se listan en la Tabla 1.

Tabla 1. Factores de pesado para los detectores PDDs.

Nivel axial del detector	Posición en pulgadas (nodo)	Nodos	Factor de pesado axial
1	6.6 (in) (28.4)	29	0.4
		28	0.6
2	8.4 (in) (26.6)	27	0.6
		26	0.4
3	12.4 (in) (22.6)	23	0.6
		22	0.4
4	22.4 (in) (12.6)	13	0.6
		12	0.4
5	26.7 (in) (8.3)	9	0.3
		8	0.7
6	28.2 (in) (6.8)	7	0.8
		6	0.2

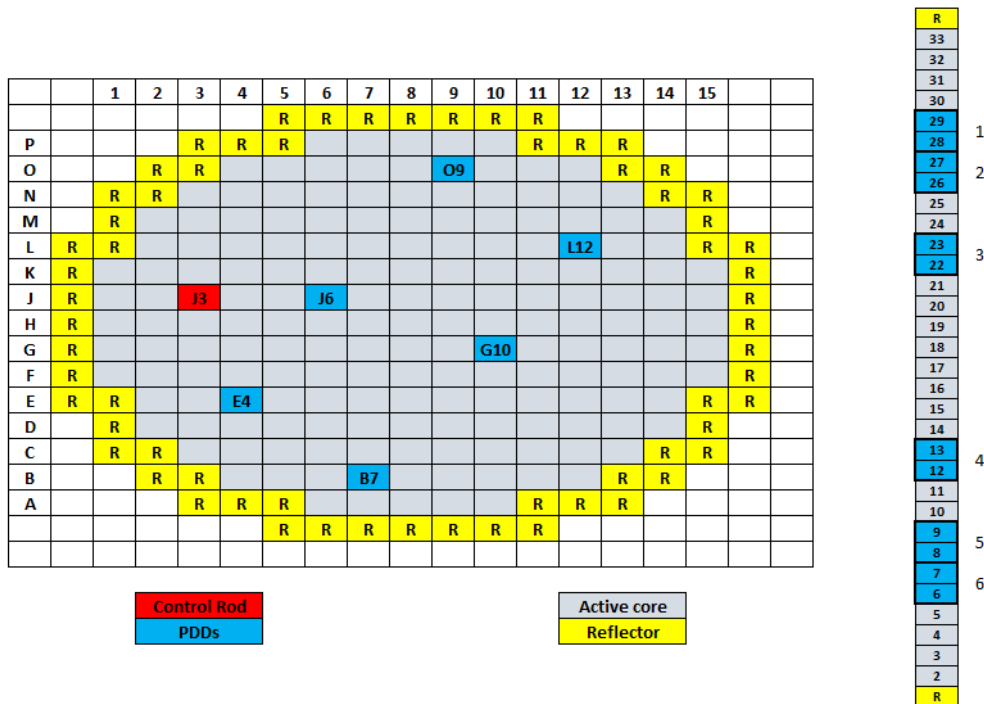


Figura 5. Localización de los PDDs en el núcleo del reactor.

6. RESULTADOS COMPUTACIONALES.

El equipo informático utilizado para las simulaciones experimentales es Quasar, un clúster que posee 4 nodos, cada uno de los cuales con procesadores AMD Opteron con 32 núcleos corriendo a 2.4 GhZ, y 96 Gb de RAM de memoria por nodo. Los nodos están conectados con Gigabit Ethernet.

6.1. Resultados del Estacionario.

En este apartado se presentan los resultados de estacionario acoplado (HFP condiciones en el BOC). Estos resultados se utilizan para verificar los archivos de entrada y las secciones eficaces antes de proceder a la simulación del transitorio de caída de barra. Para estas verificaciones se comparan los perfiles de potencia y la k_{eff} con los valores obtenidos por el código de referencia para el núcleo del reactor (REF.). En la Tabla 2 se muestra un resumen de los resultados obtenidos, los cuales confirman la validez de los modelos y secciones eficaces utilizadas.

Tabla 2. Resumen de los resultados del estacionario.

Código	Keff (-)	Error absoluto keff (pcm)	RMSE del perfil axial de potencia (%)	RMSE del perfil radial de potencia (%)
REF.	1.000141	---	---	---
PARCS alone	1.004725	458.5	2.30	1.86
pCTF/PARCS	0.999931	20.9	1.59	1.90

6.2. Resultados del Transitorio de Caída de Barra.

En este apartado se presentan los resultados del transitorio de caída de barra de control descrito con anterioridad. Durante la simulación se extrae de PARCS la información de la señal de los PDDs con la intención de comparar con las señales reales de planta. El tiempo total de simulación utilizando 5 cores del clúster Quasar es de 70 horas, lo que supone un *speedup* de 3.86 frente al caso secuencial, obteniendo una eficiencia del 77%.

En la Figura 6 se muestra la evolución de la potencia total del reactor durante la simulación comparada con la real calculada por dos métodos diferentes (señales de los detectores excore y balance término). Además en las Figuras 7 y 8 se muestran, a título ilustrativo, una comparación de la señal real de los PDDs situados en la posición J6 frente a los simulados. Como puede verse los resultados obtenidos en ambos casos son muy parecidos a las medidas adquiridas durante la prueba real.

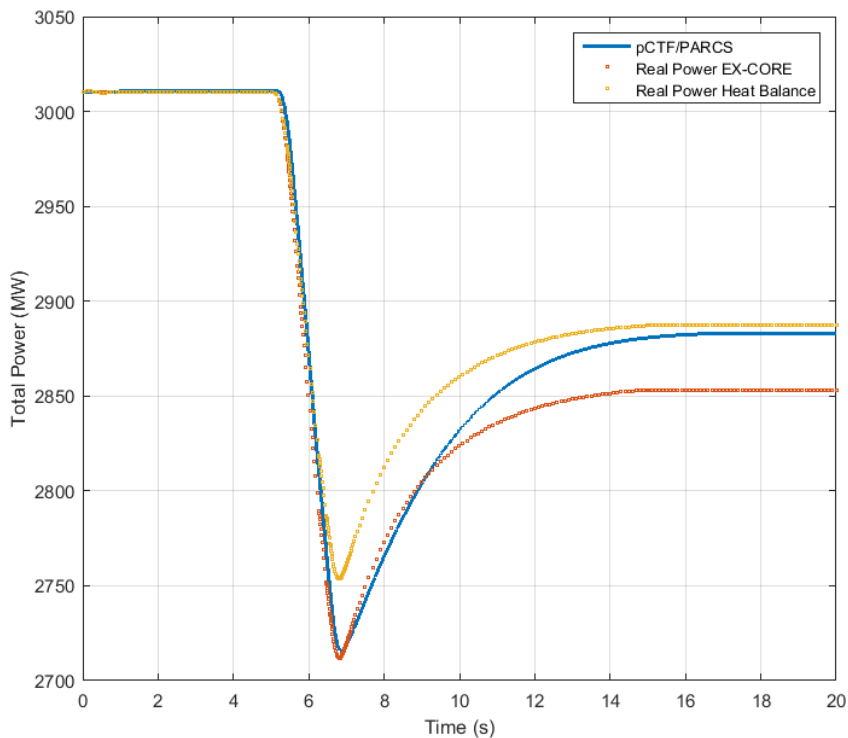


Figura 6. Potencia total del reactor durante el transitorio.

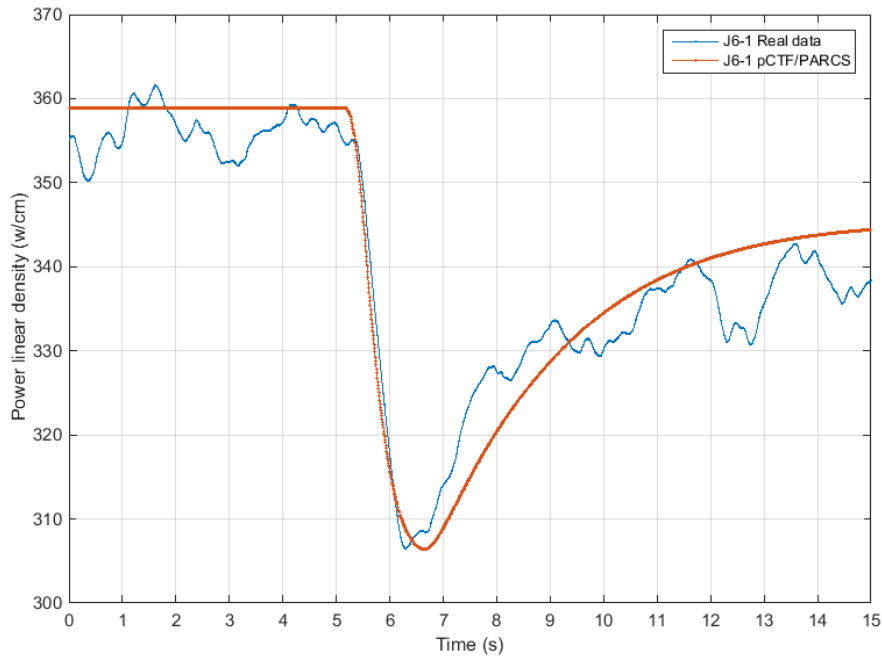


Figura 7. Comparación de la señal real y simulada para el detector situado en la posición J6-1.

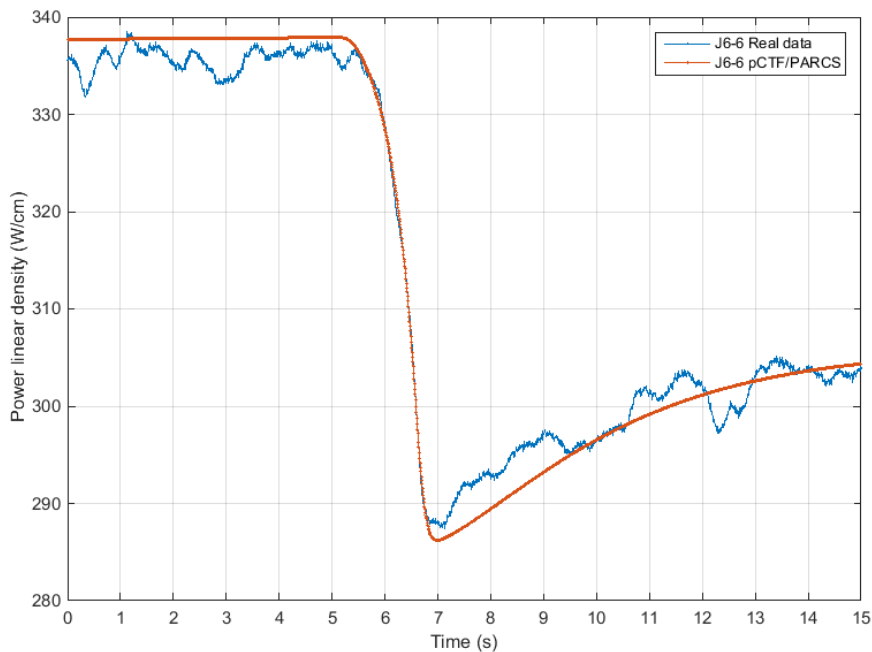


Figura 8. Comparación de la señal real y simulada para el detector situado en la posición J6-6.

7. CONCLUSIONES.

Se ha realizado un modelo a nivel de varilla completamente tridimensional del núcleo de un reactor completo de tres lazos para el código de subcanal CTF. Se han utilizado técnicas modernas de ingeniería del software así como paradigmas de computación en paralelo para crear una versión paralela de CTF, denominada pCTF, con lo que se obtiene una importante reducción del tiempo de computación cuando se simulan grandes dominios.

El código acoplado paralelo pCTF/PARCSv2.7 ha sido probado frente a un transitorio de caída de barra de control. Los resultados obtenidos de la simulación acoplada pCTF/PARCSv2.7 han sido comparados frente a los datos reales de la instrumentación de planta durante esta prueba, obteniendo señales similares en la potencia total y la lectura de los detectores PDDs durante el transitorio.

AGRADECIMIENTOS.

Resulta imprescindible destacar la colaboración de Centrales Nucleares Almaraz-Trillo (CNAT) e Iberdrola Generación Nuclear S.A. en el desarrollo del presente trabajo.

REFERENCIAS.

- [1] COBRA/TRAC, A thermal-hydraulics code for transient analysis of nuclear reactor vessels and primary coolant systems. Equations and constitutive models. vol. 1, Tech. Rep. NUREG/CR-3046, PNL-4385 (1983).
- [2] COBRA/TRAC, A thermal-hydraulics code for transient analysis of nuclear reactor vessels and primary coolant systems. User's manual. vol. 3, Tech. Rep. NUREG/CR-3046, PNL-4385 (1983).
- [3] C. Y. Paik, et al., Analysis of FLECHT-SEASET 163-rod blocked bundle data using COBRA-TF, Tech. Rep. NUREG/CR-4166, PNL (1985).
- [4] S. Balay, S. Abhyankar, M. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. C. McInnes, K. Rupp, B. Smith, H. Zhang, PETSc user's manual, Tech. Rep. ANL-95/11 - Revision 3.5, Argonne National Laboratory (2014).
- [5] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A. M. Bruaset, H. P. Langtangen (Eds.), Modern Software Tools in Scientific Computing, Birkhäuser, 1997, pp. 163–202.
- [6] MPI Forum, MPI: a message-passing interface standard, Int. J. Super-comp. Applic. High Perf. Comp. 8 (3/4) (1994) 159–416.