

ACOPLAMIENTO DE LOS CÓDIGOS TRAC-BF1/PARCSv2.7 EN LINUX SIN INTERFAZ EXTERNA DE COMUNICACIÓN

T. Barrachina¹, M. Garcia-Fenoll¹, A. Abarca¹, R. Miró¹, G. Verdú¹, A. Concejal², A. Soler³

¹*Instituto de Seguridad Industrial, Radiofísica y Medioambiental (ISIRYM).
Universitat Politècnica de València (UPV).*

*Camí de Vera, s/n.
46021 Valencia, Spain.
(tbarrachina@iqn.upv.es, mfenoll@iqn.upv.es, aabarca@iqn.upv.es,
rmiro@iqn.upv.es, gverdu@iqn.upv.es)*

²*Iberdrola Ingeniería y Construcción S.A.U.
Av. Manoteras, 20.
28050 Madrid, Spain.
(acbe@iberdrola.es)*

³*Ekergy Software S.L.
C/ Príncipe de Vergara, 34, 3º
28001 (Madrid), Spain.
(asoler@iberdrola.es)*

Resumen – *El código TRAC-BF1 sigue siendo ampliamente utilizado por la industria nuclear para análisis de seguridad. Los modelos de planta desarrollados con este código termohidráulico están sumamente validados, por lo que es aconsejable seguir mejorando este código antes de migrar completamente a otro código diferente. El acoplamiento de este código termohidráulico con el código neutrónico de la NRC PARCSv2.7 amplía las capacidades de simulación a transitorios en los que la distribución de potencia juega un papel importante. En esta ponencia se presenta el procedimiento seguido para el acoplamiento de los códigos TRAC-BF1 y PARCSv2.7 sin PVM y en Linux.*

Abstract – *The TRAC-BF1 code is still widely used by the nuclear industry for safety analysis. The plant models developed using this code are highly validated, so it is advisable to continue improving this code before migrating to another completely different code. The coupling with the NRC neutronic code PARCSv2.7 increases the simulation capabilities in transients in which the power distribution plays an important role. In this paper, the procedure for the coupling of TRAC-BF1 and PARCSv2.7 codes without PVM and in Linux is presented.*

1. INTRODUCCIÓN

El código TRAC-BF1 sigue siendo ampliamente utilizado por la industria nuclear para análisis de seguridad. Los modelos de planta desarrollados con este código termohidráulico están sumamente validados, por lo que es aconsejable seguir mejorando

este código antes de migrar completamente a otro código diferente. El acoplamiento de este código termohidráulico con el código neutrónico de la NRC PARCSv2.7 amplía las capacidades de simulación a transitorios en los que la distribución de potencia juega un papel importante.

En trabajos anteriores, los códigos TRAC-BF1 y PARCSv2.7 se acoplaron en Windows utilizando la interfaz de comunicación PVM (*Parallel Virtual Machine*). El objetivo de este trabajo es disponer de un código acoplado termohidráulico-neutrónico TRAC-BF1/PARCSv2.7 sin necesidad de utilizar ninguna interfaz de comunicación entre ambos códigos, como PVM o MPI (*Message passing interface*). Así mismo, se pretende conseguir simular cualquier transitorio utilizando el mismo ejecutable en cada una de las etapas de la simulación. Trabajar con único ejecutable del código permite un mejor control de configuración del mismo en el que se documentan todas las modificaciones y mejoras que se introducen en el código acoplado.

En esta ponencia se presenta el procedimiento seguido para el acoplamiento de los códigos TRAC-BF1 y PARCSv2.7 sin PVM y en Linux.

El artículo se organiza de la siguiente manera: en el apartado 2 se explica el criterio general seguido para el acoplamiento. El apartado 3 especifica las modificaciones llevadas a cabo en ambos códigos y finalmente, las conclusiones de este trabajo se resumen en el apartado 4.

2. ESQUEMA GENERAL DE ACOPLAMIENTO

El acoplamiento se ha abordado con la perspectiva de disponer de las dos opciones de simulación, con PVM y sin PVM. Para ello, todas las modificaciones incorporadas para el acoplamiento sin PVM se han condicionado con una opción de compilación denominada MERGEUPV. Además, en TRAC-BF1 estas modificaciones están condicionadas a que esté acoplado (mediante las variables IRUN3DK e IRPOP3D). Con esto se consigue que con el mismo ejecutable se pueda simular un caso sólo con TRAC-BF1 o acoplado con PARCS. La implementación llevada a cabo, también permite ejecutar PARCS *stand-alone* si el fichero TRACIN no existe en el directorio de trabajo. De esta forma, con el mismo ejecutable se pueden simular todas las fases de un transitorio sin utilizar la PVM.

Si no se utiliza la opción de compilación MERGEUPV se compila con PVM. De esta manera también se consigue un ejecutable que ejecuta TRAC-BF1 si el fichero TRACIN existe en el directorio de trabajo o bien ejecuta PARCS si el fichero *parcs.inp* existe en el directorio de trabajo. En este caso para simular un caso acoplado se debe tener todos los ficheros necesarios de uno y otro código en directorios diferentes (y el mismo ejecutable en ambos directorios).

La secuencia de llamadas se muestra en la Figura 1. El programa *main.f90* llama a *trac.f* con lo que TRAC-BF1 se inicia hasta llegar a la subrutina *timstp.f* donde se encuentra el primer envío de información a PARCS a través de la PVM. En este punto, se llama a *parcs.f* y éste se inicia hasta llegar a la recepción de las variables termohidráulicas (en *gi_th2neut.f* antes de llamar a *gi_comm_recvvec(TH)*). Antes de esta recepción, vuelve a TRAC-BF1, en *timstp.f* después de la llamada a *parcs.f*, de manera que continúa hasta el envío de las variables termohidráulicas (en *datato3d.f*). Después de este envío, es decir, después de llamar a *prep.f* en *trans.f*, se vuelve a PARCS que lee estas variables y continúa hasta que las vuelve a necesitar. Por tanto, PARCS ya ha escrito la potencia en la

variable *buspow_mer* y ha pasado al siguiente paso de tiempo. En la segunda vuelta de PARCS a TRAC-BF1 ya no se vuelve a *timstp.f* sino a donde se había quedado TRAC-BF1, en *trans.f* después de la llamada a *prep.f*. En este punto, el código llama a la subrutina que gestiona los controles, *cntl.f*, y vuelve a *tcal.f* para hacer el primer *edit* en el fichero OUTPUT. Para que el return de *trans.f* no sea a PARCS sino a *tcal.f* se ha añadido una llamada a *tcal.f* al final de *trans.f*. En la subrutina *tcal.f* se vuelve a llamar a TRANS, que pasa directamente a la subrutina *outer.f* (sin llamar a *timstp.f* ni a *prep.f*) y a *post.f* donde lee la potencia de PARCS. Aquí termina el primer paso de tiempo de TRAC-BF1.

A partir de este momento, en TRAC-BF1 se mantiene el bucle en *trans.f*: llama a *timstp.f*, *prep.f*, *cntl.f*, *outer.f* y *post.f*, y después de *prep.f* vuelve a *parcs.f*. PARCS recibe las variables termohidráulicas en *gi_th2neut.f* y mantiene el bucle en *parcs.f*, *dep_main.f* y *sseig.f* como se muestra en la figura.

La primera llamada desde TRAC-BF1 a PARCS se ha incluido en *timstp.f* puesto que el código ya no vuelve a pasar por este punto, sólo pasa cuando *nstep* es igual a cero.

Este esquema de acoplamiento también es válido en la simulación de un transitorio, la única diferencia está en PARCS, en el que el bucle se realiza en la subrutina *transient.f* en lugar de *sseig.h*.

El programa finaliza cuando se alcanza el tiempo final de simulación o bien cuando en estado estacionario se alcanza la convergencia en TRAC-BF1. En ambos casos, primero PARCS finaliza escribiendo correctamente la información en los ficheros de salida y antes del stop en *parcs.f*, se llama a *trac.f* para que finalice. Así es el código TRAC-BF1 el que se inicia primero y el que finaliza último.

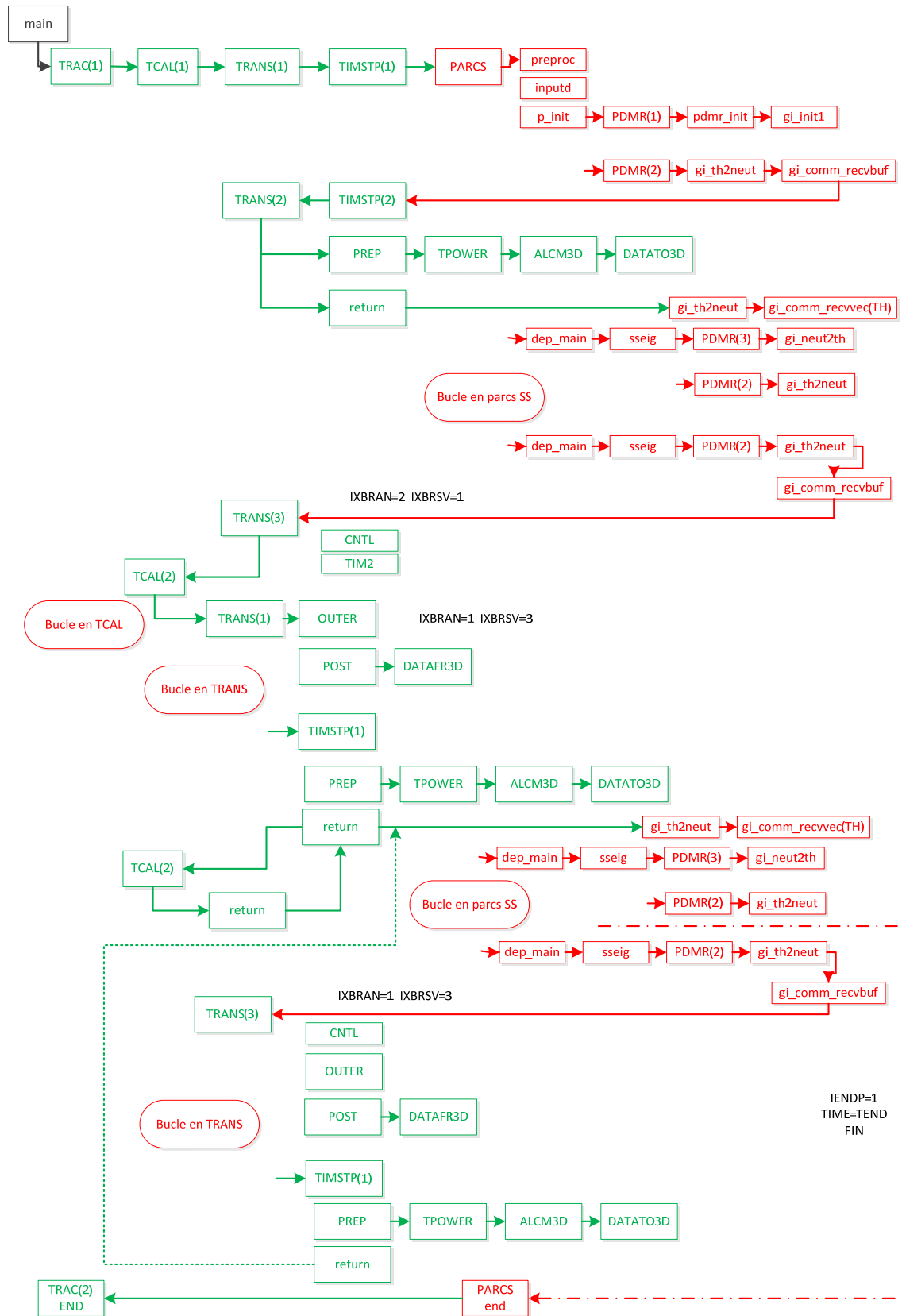


Figura 1. Esquema de acoplamiento entre TRAC-BF1 y PARCS sin PVM en estacionario.

3. MODIFICACIONES EN EL CÓDIGO FUENTE

El primer paso en el proceso de acoplamiento ha sido añadir una nueva subrutina denominada *main.f90* que controla la ejecución. En *main.f90*, el programa llama a *trac.f* si el fichero TRACIN existe en el directorio de trabajo, en caso contrario, entiende que es una ejecución de PARCS *stand alone* por lo que llama a *parcs.f* directamente. Todas las modificaciones en TRAC-BF1 están condicionadas con las variables IRPOP3D e IRUN3DK, de manera que sólo son efectivas cuando va acoplado. De esta forma se consigue poder simular todas las fases de un transitorio con el mismo ejecutable.

En ambos códigos, se añade *mergeupv.h* que contiene las variables con la información que comparten ambos códigos.

En las tablas siguientes se resumen las modificaciones llevadas a cabo en ambos códigos:

Tabla 1. Subrutinas modificadas.

TRACB	PARCS
trac.f	parcs.f
get3d01.f	gi_th2neut en gi.f
datafr3d.f	gi_exit en gi.f
datato3d.f	gi_comm_getpid en gi_commM.f
timstp.f	gi_comm_recvbuf en gi_commM.f
trans.f	gi_comm_recvvec en gi_commM.f
tcal.f	gi_comm_sendvec en gi_commM.f
aattl.f	
pol2.f	
tffm.f	
power.f	
prep.f	
post.f	
alcm3d.f	

Tabla 2. Subrutinas añadidas.

TRACB	PARCS
main.f90	
mergeupv.h	mergeupv.h

5. CONCLUSIONES

Los códigos TRAC-BF1 y PARCS v2.7 se han acoplado y compilado en entorno LINUX, con el objetivo de disponer de un único código fuente y poder llevar a cabo simulaciones sin la necesidad de la PVM. En esta ponencia se explican las modificaciones llevadas a cabo para el acoplamiento en ambos códigos.

El código acoplado se ha validado con diferentes casos. Los resultados se han comparado con la ejecución en entorno WINDOWS con ambos códigos compilados en Compaq Visual Fortran y acoplados a través de la PVM. Estos resultados se presentan en otra ponencia en este congreso.

Los resultados obtenidos con los diferentes casos analizados no hacen sospechar que el acoplamiento no sea correcto. La comparación con la simulación utilizando la PVM confirma que la información que comparten los códigos se envía y se recibe correctamente, validando así el nuevo código.

AGRADECIMIENTOS

Este trabajo ha sido llevado a cabo en colaboración con IBERDROLA INGENIERÍA Y CONSTRUCCIÓN.

REFERENCIAS

1. Idaho National Laboratory, *TRAC-BF1//MOD1: An Advanced Best Estimate Computer Program for Boiling Water Reactor Accident Analysis. Vol 2: Models and correlations*, U. S. Nuclear Regulatory Commission, Washington, DC 20555 (1992).
2. T. Downar, D. Lee, Y. Xu, T. Kozlowski, J. Staundenmier, *PARCS v2.6 U.S. NRC Core Neutronics Simulator THEORY MANUAL* (2004).