Progress in Nuclear Energy 53 (2011) 1084–1090

Contents lists available at ScienceDirect

Progress in Nuclear Energy

journal homepage: www.elsevier.com/locate/pnucene

Implementation of control rod movement and boron injection options by using control variables in RELAP5/PARCS V2.7 coupled code

R. Miró [a], T. Barrachina [a], A. Abarca [a], G. Verdú [a], C. Pereira [b,*], J.C. Martínez-Murillo [c]

[a] Institute for Industrial, Radiophysical and Environmental Safety (ISIRYM), Universitat Politècnica de València, Camí de Vera s/n, 46021 Valencia, Spain
[b] Departamento de Engenharia Nuclear, Universidade Federal de Minas Gerais, Av. Antonio Carlos 6627, 31270-901 Belo Horizonte, Brazil
[c] CNAT, Av. Manoteras, Madrid, Spain

A R T I C L E I N F O

Article history:
Received 30 May 2010
Received in revised form
5 November 2010
Accepted 23 November 2010

Keywords:
RELAP5/PARCS
Control rod
Boron injection

A B S T R A C T

In present neutron kinetics codes, control rods banks do not have the possibility of dynamic movement during the simulation of a transient; besides it is necessary to send the boron concentration from the thermal-hydraulic code to the neutronic code to account for changes in cross-sections due to boron dilution. For instance, control rod movements are pre-programmed with simple instructions introduced before the beginning of the calculation. Hence, control rod positions are not related to the core characteristics and the control systems at any time of the simulation. This work presents the changes introduced in RELAP5/PARCS v2.7 codes to achieve that control rods and the boron injection become more dynamic and realistic components in such kind of simulators. Furthermore, in order to test the modifications introduced in both codes, it has been analyzed a boron injection transient in a typical PWR Nuclear Power Plant. The thermal-hydraulic model includes all the primary loop components of a PWR, the core fuel assemblies modeled with *PIPE* components, pumps, steam generators, pressurizer, etc. The neutronic representation of the reactor has been made in a one-to-one basis fuel channel model for the whole core.

1. Introduction

In order to efficiently characterize realistic transients, as the Reactivity Insertion Accidents (RIA), using coupled neutronic-thermal-hydraulic 3D best estimate system codes, like RELAP5/PARCS v2.7 coupled code (Downar et al., 2006; RELAP/MOD3.3 Code Manual Volume VIII: Programmers Manual, 2001), it is necessary to introduce some improvements in simulations by adding the capability of control rod movement and boron injection by means of RELAP5 control variables, with the aim of being able to analyze dynamically asymmetric transient accidents in a nuclear power reactor, reproducing all control systems present in commercial reactors.

In present neutron kinetics codes, control rods banks do not have the possibility of dynamic movement during the simulation of a transient. For instance, control rod movements are pre-programmed with simple instructions introduced before the beginning of the calculation. Hence, control rod positions are not related to the core characteristics and the control systems at any time of the simulation. Besides, it is necessary to send the boron concentration from the thermal-hydraulic code to the neutronic code to account for changes in cross-sections due to boron dilution.

This work presents the changes introduced in RELAP5/PARCS v2.7 codes to achieve that control rods and the boron injection become more dynamic and realistic components in such kind of simulators.

With these modifications, control rods can be moved automatically, activated by the RELAP5 code control system, and they can depend on signals related to the reactor dynamics, like pressure, fuel temperature or moderator temperature, etc., improving the realism of the calculation and widening the simulation possibilities. RELAP5 calculates the boron concentration in each node of the channels representing the reactor core, sending this information to the PARCS neutronic code.

The environment chosen for this work was the Compaq Visual Fortran 6.6A (CVF 6.6A) (Compaq Visual Fortran, 2001) graphical programming environment. The fundamental reasons were, on one hand the ease of coding and versatility of the debugger of this environment, and on the other hand, the fact that the PARCS v2.7 source code came already prepared for this environment. RELAP5 did not come prepared for this graphical environment of

* Corresponding author.
E-mail addresses: rmiro@iqn.upv.es (R. Miró), tebarcel@upvnet.upv.es (T. Barrachina), gverdu@iqn.upv.es (G. Verdú), claubia@nuclear.ufmg.br (C. Pereira), jcmm@cnat.es (J.C. Martínez-Murillo).

0149-1970/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.pnucene.2010.11.001

programming: originally its compilation was made with UNIX (or CYGWIN) “shell scripts” files, being all the process hidden for the user. Therefore, it was necessary to adapt the source code of RELAP5 to CVF 6.6A.

As a practical qualification, a transient for a PWR NPP is presented and results are compared using both methodologies. Furthermore, it analyzes a boron injection transient in a typical PWR Nuclear Power Plant. The thermal-hydraulic model includes all the primary loop components of a PWR, the core fuel channels modeled with *PIPE* components, pumps, steam generators, pressurizer, etc. The neutronic representation of the reactor has been made in a one-to-one basis fuel channel model for the whole core.

2. Implementation of control rod movement

The criteria adopted in order to approach the implementation of the codes modifications were the following:

- The control variables will be used to move the control rod banks of the reactor core.
- The control variables ID’s have to be introduced in the MAPTAB file, using the %CRSIG card.
- The control variables ID’s will be introduced by order, so that first it will move the first control rod bank and so on. If a position is left with a “0” value, this will indicate that control bank will not be moved, and therefore it will maintain its initial position during the whole transient.
- Each control rod bank will be controlled only by a control variable. For each time step the position of the control rod bank will be equal to the value of the control variable in that time step.
- The user can opt for moving the control rods with the usual method or using this new option, although this option dominates the previous one. Thus, if the user implements the card % CRSIG in the MAPTAB file, the control rods will be moved using the new option, although in the PARCS input, the orders of movement are also being implemented (as in the normal case up to now).
- It is recommendable that the initial position of the control rods corresponds with the initial position indicated in the PARCS input file (Fig. 1).

The variables that have been introduced in the modification of the subroutines are summarized in Table 1. Table 2 shows a summary of the both codes subroutines that have been modified:

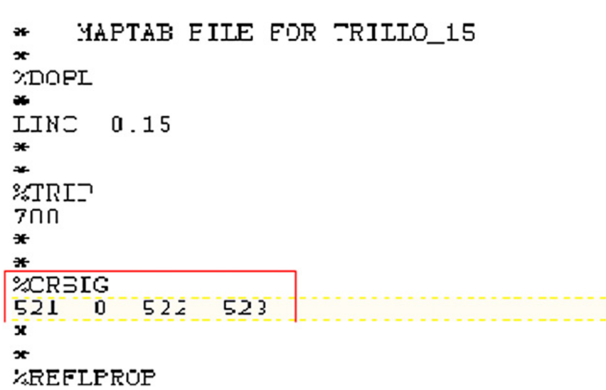

Fig. 1. Data entry in MAPTAB file.

Table 1
Summary of the new variables declaration in the PARCS V2.7 and RELAP5 codes.

Variable	Code	File
n_var	PARCS v2.7	pdmr_varM.f
var_leidas(1000)	PARCS v2.7	pdmr_varM.f
mov_cr_relap	PARCS v2.7	pdmr_varM.f
var_cntrl(1000)	RELAP5	r-var.f
guardar_cntrlv(1000)	RELAP5	r-var.f
contador	RELAP5	r-commu.f
contador2	RELAP5	r-commu.f

Table 2
Summary of the modified subroutines in PARCS V2.7 and RELAP5 codes.

File	Modification	Code
pdmr_varM.f	Declaration of the new variables: var_leidas, n_var, mov_cr_relap	PARCS v2.7
pdmr_mapM.f	New loop to see the information from the card %CRSIG	PARCS v2.7
gi_commM.f	Variables var_leidas y n_var are sent and variable var_cntrl is received through PVM (PVM,1994)	PARCS v2.7
perturb.f	With the variable mov_cr_relap, control rods are moved following the RELAP5 instructions	PARCS v2.7
r-var.f	Declaration of the new variables: var_leidas, n_var, guardar_cntrlv, var_cntrl	RELAP5
r-commu.f	The variables var_cntrl is sent and the variables var_leidas and n_var are receive through PVM	RELAP5
iconvr.f	Stores the initial values of the variable guardar_cntrlv	RELAP5
tran.f	First it updates the values and then it sends them through PVM (first it calls to subroutine convar and then calls to subroutine rdmr)	RELAP5

2.1. Qualification of the modifications introduced

In order to test that the modifications have been introduced properly, it has been analyzed a transient in a PWR NPP, comparing the results obtained moving the control rods using the original options in PARCS input file with the results obtained moving the control rods using the new control variables introduced.

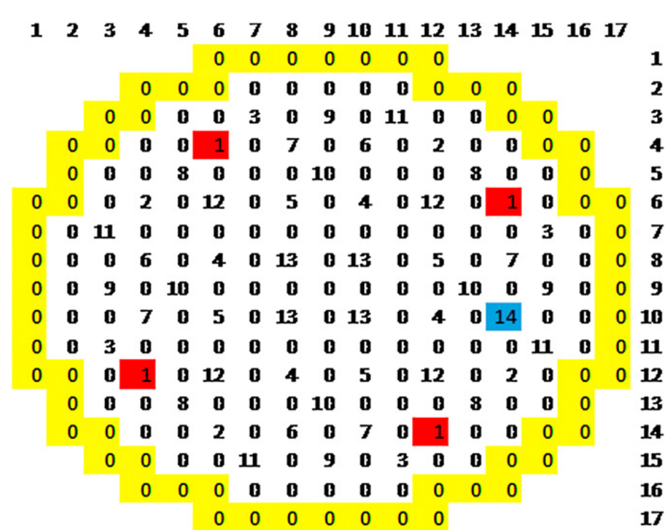

Fig. 2. Control rod banks moved.