SRAC for Windows OS

User Manual and Guide

Introduction

SRAC (Standard Reactor Analysis Code) is a set of FORTRAN computer codes and routines made by JAERI for analyzing various reactor model. It utilizes Linux terminal and c-shell script engine to run on personal computer, an unfamiliar way to operate computer for novel users nowadays. Due to different architecture between Linux and Windows operating systems, users tend to install Linux OS in order to be able to run SRAC. This approach doesn't come handy and limits the spread of SRAC usage even in academic and research environment (at least in Indonesia where most people use Windows operating system).

We present SRAC as compiled binaries that works out of the box in Windows OS. You don't need to install anything else to run SRAC (from now on, SRAC refers to SRAC for Windows). Input data and various other commands are written in PowerShell scripts (an equivalent of c-shell scripts in Windows). This manual will guide you to install and run SRAC in your Windows PC.

Prerequisites

You'll need latest Microsoft Visual C++ Runtime, download it from Microsoft website. I used Visual C++ Compiler 2012 to compile SRAC, so Visual C++ Runtime 2012 should be enough. It's used by many software, so your computer is most likely installed them already.

Getting Started

First of all, you need to obtain SRAC binaries from internet (right now we only distribute SRAC via Facebook). Extract it to your desired folder, and you're done. For convenience, we recommend to extract SRAC on top level directory (C:\, D:\, E:\). Here's SRAC directory would look like:

```
D:\SRACW
    SRAC
        bin
           SRAC.exe
        cmnd
        .
        .
        .
        smpl
           outp
           shr
        .
        .
        .
    SRAC Manual
    SRACLIB-EDF70
        .\PunchMe.ps1
    SRACLIB-JDL33
        .\PunchMe.ps1
```

Next, you'll need to obtain nuclear cross-section libraries so SRAC can use those data while simulating your reactor model. Our SRAC package should include two libraries, you can use them or use another library. These libraries would have to be decompressed first. Find **PunchMe.ps1** at library folder, right-click to open context menu and choose **Run with PowerShell**. Sit back and relax since it will take a while. After the process is done, your library folder size will increase substantially. Congratulation, you've installed SRAC in your computer. Seasoned SRAC users from Linux shall find many hassle is skipped here.

Now let's check whether everything is up and running. Find **smpl** directory in SRAC folder (..\SRACW\SRAC\smpl) and look inside **shr** folder, you'll find various PowerShell scripts for various reactor model there. Let's check **Test.ps1** (some portions of text are omitted):

```
#===============================================================================
#  Test.ps1 : Test problem to check SRAC installation
#            UO2 pin cell calculation in LWR next generation
#            fuel benchmark (No burn-up)
#  Options : Pij(Geometry type IGT=4), PEACO
#===============================================================================

$SRAC_DIR = "E:\SRACs\SRAC"
$LMN = "SRAC.exe"
$BRN = "u4cm6fp50bp16T"
$ODR = "$SRAC_DIR\smpl\outp"
$Case = "Test"
$PDSD = "$SRAC_DIR\tmp"

#
#=============  mkdir for PDS  =================================================
#
#  PDS_DIR : directory name of PDS files
#  PDS file names must be identical with those in input data
#
$PDS_DIR = "$PDSD\$Case"
mkdir "$PDS_DIR"
mkdir "$PDS_DIR\UFAST"
mkdir "$PDS_DIR\UTHERMAL"
mkdir "$PDS_DIR\UMCROSS"
mkdir "$PDS_DIR\MACROWRK"
mkdir "$PDS_DIR\MACRO"
mkdir "$PDS_DIR\FLUX"
mkdir "$PDS_DIR\MICREF"

#
#=============  Change if you like =============================================
#

$LM = "$SRAC_DIR/bin/$LMN"
$DATE = Get-Date -uformat "%Y.%m.%d.%H.%M.%S"
$WKDR = "$SRAC_DIR/SRACtmp.$Case.date $Date"
mkdir $WKDR
#
#-- File allocation
#  fu89 is used in any plot options, fu98 is used in the burnup option
#  Add other units if you would like to keep necessary files.
   $env:fu50 =  "$SRAC_DIR/lib/burnlibT/$BRN"
   $env:fu85 =  "$SRAC_DIR/lib/kintab.dat"
#  $env:fu89 =  "$ODR/$CASE.SFT89.$DATE.SAMPLE"
#  $env:fu98 =  "$ODR/$CASE.SFT98.$DATE.SAMPLE"
   $env:fu99 =  "$ODR/$CASE.SFT99.$DATE.SAMPLE"
   $OUTLST   =  "$ODR/$CASE.SFT06.$DATE.SAMPLE"

#
#=============  Exec SRAC code with the following input data ===================
#
```

```
cd $WKDR
$Input = @"
TEST : Case name (A4)
UO2 pin cell problem in LWR next generation fuel benchmark (No burn-up)
************************************************************************
*  Benchmark Reference :
*  A.Yamamoto, T.Ikehara, T.Ito, and E.Saji : "Benchmark Problem for
*  Reactor Physics Study of LWR Next Generation Fuels",
*  J. Nucl. Sci. Technol., Vol.39, No.8, pp.900-912, (2002).
************************************************************************
1 1 1 1 2   1 4 3 -2 1   0 0 0 0 1   2 1 0 0 0 / SRAC CONTROL
1.000E-20 / Geometrical buckling for P1/B1 calculation
*- PDS files ------2---------3---------4---------5---------6---------7--
* Note : All input line must be written in 72 columns except comments
*        even when environmental variables are expanded.
E:/SRACs/SRACLIB-EDF70/pds/pfast   Old  File
E:/SRACs/SRACLIB-EDF70/pds/pthml   O    F
E:/SRACs/SRACLIB-EDF70/pds/pmcrs   O    F
$PDS_DIR/UFAST      Scratch  Core
$PDS_DIR/UTHERMAL   S        C
$PDS_DIR/UMCROSS    S        C
$PDS_DIR/MACROWRK   S        C
$PDS_DIR/MACRO      S        C
$PDS_DIR/FLUX       S        C
$PDS_DIR/MICREF     S        C
************************************************************************
62 45  8 8 /  107 group => 16 group
62(1)      /  Energy group structure suggested for LWR analyses
45(1)      /
8 11  9  9  9  9  4  3  / Fast 8 group
3  3  3  6  3 10  9  8  / Thermal 8 group
***** Enter one blank line after input for energy group structure

***** Input for PIJ (Collision Probability Method)
4 7 7 3 1   1 7 0 0 0   5 0 6 15 0   0 45 0         / Pij Control
0 50 50 5 5 5 -1  0.0001 0.00001 0.001 1.0 10. 0.5  /
1 1 1 2 3 3 3  /  R-S
3(1)           /  X-R
1 2 3          /  M-R
0.0  0.238  0.336  0.412  0.476  0.528  0.580  0.6325    / RX
****** Input for material specification
3 / NMAT
FUELX01X  0 3  900.0  0.824    0.0  / 1 : UO2 fuel
XU050000  2 0  1.5122E-03      /1
XU080000  2 0  2.1477E-02      /2
XO060000  0 0  4.5945E-02      /3
CLADX02X  0 1  600.0  0.128    0.0  / 2 : cladding
XZRN0000  2 0  4.3107E-02      /1
MODEX03X  0 2  600.0  1.0      0.0  / 3 : moderator
XH01H000  0 0  4.4148E-02      /1
XO060000  0 0  2.2074E-02      /2
****** Input for cell burn-up calculation (when IC20=1)
*  31 1 1 0 0  0 0 0 0 0  10(0)   / IBC
*  31(1.790E-04)   / Power level  (MWt/cm)
*  0.10E+03  1.00E+03  2.50E+03  5.00E+03  7.50E+03  1.00E+04
*  1.25E+04  1.50E+04  1.75E+04  2.00E+04  2.25E+04  2.50E+04
*  2.75E+04  3.00E+04  3.25E+04  3.50E+04  3.75E+04  4.00E+04
*  4.25E+04  4.50E+04  4.75E+04  5.00E+04  5.25E+04  5.50E+04
*  5.75E+04  6.00E+04  6.25E+04  6.50E+04  6.75E+04  7.00E+04
*  7.25E+04  /  keff calculation is not done at the last step
****** Input for PEACO option
0    / no plot
****** Enter one blank line to terminate repeatation on calculation cases


"@

$Input | &"$LM" >> $OUTLST
```

In order to work properly, the script needs to know about SRAC executables and nuclear cross-section libraries whereabouts. You have to point correct directory in your computer by editing text marked with red font above. Green text above is called **SRAC free-form input**, you put your reactor model there according to SRAC free-form input format which can be found on SRAC Manual. **Note that directory in SRAC free-form input uses forward-slash ('/') as directory level separator.** We won't delve into SRAC free-form input here, please consult SRAC Manual by yourself.

Now let's check whether SRAC is working properly. Run **Test.ps1** with PowerShell and wait for a while (it should only takes brief moment). After the console is closed, look inside **outp** folder, you'll find SRAC calculation result for this case there. In order to check whether something's gone wrong, compare your output file with older output file which already resides there. If you don't find anything strange then it works. Feel free to check other samples.

Advanced Topics

Building SRAC for Windows

It is rather unfortunate that while building SRAC for Windows is not hard, it is not quite straight-forward. This prevents me to provide scripts and makefiles that were used to build SRAC for Windows in the package, as you'll most likely stumble across problems when use them. If you're really interested though, feel free to mail me on lebenasa(at)gmail(dot)com.

Run SRAC with Multi-Processor

Right now, SRAC is capped to run on single processor. Compiler flag doesn't help here, so SRAC source code itself need to be altered to use all available processor in calculation. This optimization would lead to multiple decrease in computing time, a really useful improvement. If you're familiar with FORTRAN and C and interested in nuclear reactor simulation, why don't you try to optimize SRAC? I'd be more than glad to help.

Acknowledgements

SRAC is propriety right of JAERI. This work doesn't change anything in original source code.