Writing Benchmark Models with a Computer Program

Jeffrey A. Favorite
Radiation Transport Applications Group (XTD-RTA)
Los Alamos National Laboratory

Nuclear Criticality Safety Program Technical Program Review
Riverhead, NY
February 20–22, 2024
Introduction

• The goal is to automate as much as possible.

• The program is a tool for a specific evaluation.

• I am not presenting or promoting a generic computer program for all benchmark evaluations!

• I’ve been doing this since my first evaluation in 2012.

• There was a recent talk along similar lines:
  Transactions of the American Nuclear Society, 127, 678–681 (2022);
  ANS Winter Meeting, Phoenix, AZ.

• The purposes of this talk:
  1. Encourage efficiency!
  2. Find out how you do this.

• Caveat: No tool or algorithm is foolproof or solves all problems.
Making a Model—Part I

- Set dimensions
- Set masses
- Calculate volumes
- Calculate part mass densities
- Calculate material and cell atom densities

```fortran
  call set_dims_dwg(maxparts,maxdims,maxangs,dims,angs)
  if(i Mos_dim == 1)then
    call set_dims_Mos(maxparts,maxdims,maxangs,dims,angs)
  end if
  if(i meas_dim == 1)then
    call set_dims_meas(maxparts,maxdims,maxangs,dims,angs)
  end if
  if(i ta18_dim == 1)then
    call set_dims_ta18(maxparts,maxdims,maxangs,dims,angs)
  end if
  dims(1:maxparts,1:maxdims)=dims(1:maxparts,1:maxdims)*in2cm
  call calc_volume(maxparts,maxdims,maxangs,i_meas_dim,pname,dims,angs,volume)
  call set_mass(maxparts,i_ta18_dim,pname,mass)
  call calc_dens(maxparts,pname,volume,mass,dens)
  call calc_atdens(iuo,1,denmat,den_calc)

  call set_dims_meas(maxparts,maxdims,maxangs,dims,angs)
  end if
```

- Write the cell densities with the cells.

If this only had to be done once, automation would not be necessary.

Measurement Worksheets (notes for Jul-11-22)

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass(R01)</td>
<td>8381.7d0</td>
</tr>
<tr>
<td>mass(R02)</td>
<td>8131.4d0</td>
</tr>
<tr>
<td>mass(R03)</td>
<td>8022.6d0</td>
</tr>
<tr>
<td>mass(R04)</td>
<td>6880.7d0</td>
</tr>
<tr>
<td>mass(R05)</td>
<td>8155.5d0</td>
</tr>
<tr>
<td>mass(R06)</td>
<td>9282.4d0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ring 101 (128Y1719423)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dims(R101.1)</td>
</tr>
<tr>
<td>dims(R101.2)</td>
</tr>
<tr>
<td>dims(R101.3)</td>
</tr>
<tr>
<td>dims(R101.4)</td>
</tr>
<tr>
<td>dims(R101.5)</td>
</tr>
<tr>
<td>dims(R101.6)</td>
</tr>
<tr>
<td>dims(R101.7)</td>
</tr>
<tr>
<td>dims(R101.8)</td>
</tr>
<tr>
<td>dims(R101.9)</td>
</tr>
<tr>
<td>dims(R101.10)</td>
</tr>
<tr>
<td>dims(R101.11)</td>
</tr>
<tr>
<td>dims(R101.12)</td>
</tr>
<tr>
<td>angs(R101.1)</td>
</tr>
<tr>
<td>angs(R101.2)</td>
</tr>
</tbody>
</table>
Making a Model—Part II

- Set surface locations and dimensions.
- Write surfaces and their parameters.

For this one I made a list of parts in the stack:

c list_refl(1:max_refl_cells)=0
c list_core(1:max_core_cells)=0
c list_out(1:max_out_cells)=0

c parts in reflector and Planet
n=0
n+n+1 : list_refl(n)=Tilt_Weigh_Table
n+n+1 : list_refl(n)=Platen_Adapter_Plate
n+n+1 : list_refl(n)=Platen_HeatSink
n+n+1 : list_refl(n)=Bottom_Reflector
n+n+1 : list_refl(n)=SR_Base

select case (ic)
case (1)
  n+n+1 : list_refl(n)=SR_Middle_030_1
  n+n+1 : list_refl(n)=SR_Middle_030_2
  n+n+1 : list_refl(n)=SR_Middle_030_3
  n+n+1 : list_refl(n)=SR_Flat_Upper_030
  n+n+1 : list_refl(n)=Core_Support_Plate
  n+n+1 : list_refl(n)=SR_Flat_Lower_030
  n+n+1 : list_refl(n)=SR_Middle_030_4
  n+n+1 : list_refl(n)=SR_Middle_030_5
  n+n+1 : list_refl(n)=SR_Trans_030
  n+n+1 : list_refl(n)=TR_1000_1
  n+n+1 : list_refl(n)=TR_1000_2
  n+n+1 : list_refl(n)=TR_1000_3
  n+n+1 : list_refl(n)=RTD_slot_r

  c block 2. surfaces
  else if(nb.eq.2)then
    write(iuo,`(""`) ! blank
    nsurf=0
    zzz1=zbase
  c reflector surfaces
  do n=1,nrefl
    c write(x,`("2i6")`)n.list_refl(n)
    if(list_refl(n).lt.0)then
      cycle
    else if(list_refl(n).eq.Tilt_Weigh_Table)then
      write(iuo,`("c ",a)`trim(pname(list_refl(n))))
      nsurf=nsurf+1
      zzz2=zzz1+dimsp(list_refl(n),3)
      xxx=dimsp(list_refl(n),1)/2.0
      yyy=dimsp(list_refl(n),2)/2.0
      call write_surf(nsurf,iuo)
      write(iuo,`("rpp",5x,0pF11.6")`) 1

  xxx,xxx,yyy,yyy,zzz1,zzz2
    if(ic.eq.1)then
      core_stack(4,1)=zzz2
      refl_stack(1,1)=zzz2
    else if(ic.eq.2)then
      core_stack(10,1)=zzz2
      refl_stack(3,1)=zzz2
    end if
  call write3(1,iuo,pname(list_refl(n)),zzz1,zzz2)
  xxx=dimsp(list_refl(n),11)/2.0
  yyy=dimsp(list_refl(n),12)/2.0
  rrr=dimsp(list_refl(n),13)/2.0
  do i=1,4
    nsurf=nsurf+1
    call write_surf(nsurf,iuo)
    write(iuo,`("c/z",5x,a,0pF9.6,a,F9.6,F11.6")`) 1
    c4(1,i),xxx,c4(2,i),yyy,rrr
  end do ! 1
  zzz1=zzz2
### Auxiliary Files

- Need a base or “skeleton” input file.
  - It is easier for a code to alter something than to create from scratch.
  - Probably the base input should have cells defined, but the densities will be written each time.
  - Different surface arrangements can be accommodated with if/then structures.

- Material files  
  Ex. SS 303:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>c SS 303 (8.0 g/cc)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>0.075</td>
<td>c begin at.dens., wt.dens. 8.829956206E-02 8.0000000000000</td>
</tr>
<tr>
<td>Si</td>
<td>1.00</td>
<td>c round at.dens., wt.dens. 8.829894685E-02 7.999944324017</td>
</tr>
<tr>
<td>P</td>
<td>0.10</td>
<td>s1 nlib=00c</td>
</tr>
<tr>
<td>S</td>
<td>0.30</td>
<td>6012 2.9752E-04 6013 3.3091E-06</td>
</tr>
<tr>
<td>Cr</td>
<td>18.00</td>
<td>14028 1.5821E-03 14029 8.0107E-05</td>
</tr>
<tr>
<td>Mn</td>
<td>1.00</td>
<td>14030 5.3176E-05</td>
</tr>
<tr>
<td>Fe</td>
<td>70.225</td>
<td>15031 1.5554E-04</td>
</tr>
<tr>
<td>Ni</td>
<td>9.00</td>
<td>16032 4.2830E-04 16033 3.3806E-06</td>
</tr>
<tr>
<td>Mo</td>
<td>0.30</td>
<td>16034 1.8977E-05 16036 9.0150E-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24050 7.2465E-04 24052 1.3974E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24053 1.5844E-03 24054 3.9443E-04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25055 8.7693E-04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>26054 3.5744E-03 26056 5.5567E-02</td>
</tr>
<tr>
<td></td>
<td></td>
<td>26057 1.2723E-03 26058 1.6963E-04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>28058 5.0488E-03 28060 1.9283E-03</td>
</tr>
<tr>
<td></td>
<td></td>
<td>28061 8.3485E-05 28062 2.6523E-04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>28064 6.7232E-05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>42092 2.2358E-05 42094 1.3936E-05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>42095 2.3985E-05 42096 2.5130E-05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>42097 1.4388E-05 42098 3.6354E-05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>42100 1.4509E-05</td>
</tr>
</tbody>
</table>

*Materials can be defined within the code.*
My System is Based on Parts and Materials

<table>
<thead>
<tr>
<th>Part Indices</th>
<th>Material Indices</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITISP-1</td>
<td>SS303-1</td>
</tr>
<tr>
<td>UISP-IITISP+</td>
<td>SAE4340-SS303+1</td>
</tr>
<tr>
<td>SafetyBlock-UISP+</td>
<td>Vasco-SAE4340+1</td>
</tr>
<tr>
<td>R101-SafetyBlock+</td>
<td>Al10601-Vasco+1</td>
</tr>
<tr>
<td>R102-R101+</td>
<td>Lex-Al10601+1</td>
</tr>
<tr>
<td>R103-R102+</td>
<td>AISI1019-Lex+1</td>
</tr>
<tr>
<td>R104-R103+</td>
<td>mR101-AISI1019+1</td>
</tr>
<tr>
<td>R105-R104+</td>
<td>mR102-mR101+1</td>
</tr>
<tr>
<td>R106-R105+</td>
<td>mR103-mR102+1</td>
</tr>
<tr>
<td>CR1-R106+</td>
<td>mR104-mR103+1</td>
</tr>
<tr>
<td>CR2-CR1+</td>
<td>mR105-mR104+1</td>
</tr>
<tr>
<td>BR-CR2+</td>
<td>mR106-mR105+1</td>
</tr>
<tr>
<td>Spindle-BR+1</td>
<td>mSafetyBlock-R106+1</td>
</tr>
<tr>
<td>AlignPin-Spindle+1</td>
<td>UISP-mSafetyBlock+1</td>
</tr>
<tr>
<td>SafetyBase-AlignPin+1</td>
<td>mCR1-mUISP+1</td>
</tr>
<tr>
<td>Pads-SafetyBase+1</td>
<td>mCR2-mCR1+1</td>
</tr>
<tr>
<td>BRing-Pads+1</td>
<td>mBR-mCR2+1</td>
</tr>
<tr>
<td>SubCvrP1-BRing+1</td>
<td>Room=98</td>
</tr>
<tr>
<td>MountP1-SubCvrP1+1</td>
<td>mAir=99</td>
</tr>
<tr>
<td>Leg-MountP1+1</td>
<td>maAir=mAir</td>
</tr>
<tr>
<td>Clamp-Leg+1</td>
<td>denmat(SS303)=8.0d</td>
</tr>
<tr>
<td>BellyBand-Clamp+1</td>
<td>denomat(SS303)=8.0d</td>
</tr>
<tr>
<td>NutsBolts-BellyBand+1</td>
<td>denomat(SAE4340)=7.85d0</td>
</tr>
<tr>
<td>CrCvr-NutsBolts+1</td>
<td>denomat(Vasco)=8.0d</td>
</tr>
<tr>
<td>CntSh-CrCvr+1</td>
<td>denomat(AL10601)=2.70d</td>
</tr>
<tr>
<td>MntP12-CntSh+1</td>
<td>if(i_Moss_den == 0)denomat(Lex)=1.20d0</td>
</tr>
<tr>
<td>Room-MntP12+1</td>
<td>denomat(AISI1019)=7.87d0</td>
</tr>
<tr>
<td>Air-Room+1</td>
<td>denomat(mAir)=1.052d-3</td>
</tr>
</tbody>
</table>
Geometry Perturbations

- This is where this system really pays off!

- I have used a separate code for each type of part (e.g., fuel, reflector, structure) or evaluation subsection.

Will go on to calculate volumes and densities using perturbed dimensions

- Volumes are not always analytic.

```plaintext
parameter(nparts=4)

data parts/
  1 "Plate_", "StudL_", "StudB_", "Spacer_"/
data ndims/
  1 2, 2, 2, 3 /

c reset everything
dimsp(1:maxparts,1:maxdims)=dims(1:maxparts,1:maxdims)
volume_p(1:maxparts)=volume(1:maxparts)
den_p(1:maxparts)=den(1:maxparts)
devn=0.d0
c 1 "Plate_", "StudL_", "StudB_", "Spacer_"/
c 1 2, 2, 2, 3 /
c make perturbations, dimensions are in cm here.
select case(ifl)
case (1)
 if(id.eq.1)then
dimsc="lw_"
devn=0.030d 1 inches
dr=rp(ip1)*devn*in2cm
 dimsp(Compress Plate.1)=dimsp(Compression Plate.1)+dr
dimsp(Compress Plate.2)=dimsp(Compression Plate.2)+dr
 end if
case (2)
 if(id.eq.1)then
dimsc="th_"
devn=0.030d 1 inches
dr=rp(ip1)*devn*in2cm
 dimsp(Compress Plate.3)=dimsp(Compression Plate.3)+dr
case (3)
 if(id.eq.1)then
dimsc="ht_"
devn=0.030d 1 inches
dr=rp(ip1)*devn*in2cm
 dimsp(Compress Stud Top.3)=dimsp(Compression Stud Top.3)+dr
case (4)
 if(id.eq.1)then
dimsc="dd_"
devn=0.030d 1 inches
dr=rp(ip1)*devn*in2cm
 dimsp(Compress Stud Top.4)=dimsp(Compression Stud Top.4)+dr
```
```plaintext
dims(Mod_Inn_600_01:Mod_Inn_600_18,3:4)=
dims(Mod_Inn_600_01:Mod_Inn_600_18,3:4)
2 *(1.d0+alphpat(CPYC_600)*devn) ! z, outer diameter

c
dims(ZPPR_Sleeve,1:4)=dims(ZPPR_Sleeve,1:4)
2 *(1.d0+alphpat(Cladding)*devn) ! x, y, z, thickness
dims(ZPPR_Plug,1)=dims(ZPPR_Plug,1)
2 *(1.d0+alphpat(SS304L)*devn) ! x
dims(ZPPR_Spring,1:3)=dims(ZPPR_Spring,1:3)
2 *(1.d0+alphpat(Carbon_Steel)*devn) ! x, y, z

c warning. using one value for all.
dims(ZPPR_PANN_A:ZPPR_PANN_S,1:3)=
dims(ZPPR_PANN_A:ZPPR_PANN_S,1:3)
2 *(1.d0+alphpat(PANN_A)*devn) ! x, y, z

c

Los Alamos
NATIONAL LABORATORY
UNCLASSIFIED

Operated by Triad National Security, LLC for the U.S. Department of Energy's NNSA

Slide 8 of 12

Operated by Triad National Security, LLC for the U.S. Department of Energy's NNSA

NNSA
Thermal Expansion Calculations Are Easy

- The change $\Delta x$ in dimension $x$ due to a change in temperature $\Delta T$ is $\Delta x = x' - x_0 = \Delta T \alpha_x x_0$
  + Or $x' = x_0 (1 + \alpha_x \Delta T)$

```plaintext
dims(Mod_Inn_600_01:Mod_Inn_600_18,3:4)=
dims(ZPRR_Sleeve,1:4)=dims(ZPRR_Sleeve,1:4)
2 *(1.d0+alphat(EPYC_600)*devn) ! z, outer diameter

c
dims(ZPRR_Sleeve,1:4)=dims(ZPRR_Sleeve,1:4)
2 *(1.d0+alphat(Cladding)*devn) ! x, y, z, thickness
dims(ZPRR_Plug,1)=dims(ZPRR_Plug,1)
2 *(1.d0+alphat(SS304L)*devn) ! x
dims(ZPRR_Spring,1:3)=dims(ZPRR_Spring,1:3)
2 *(1.d0+alphat(Carbon_Steel)*devn) ! x, y, z

Warning, using one value for all.
dims(ZPRR_PANN_A:ZPRR_PANN_S,1:3)=
1 dims(ZPRR_PANN_A:ZPRR_PANN_S,1:3)
2 *(1.d0+alphat(PANN_A)*devn) ! x, y, z
```

\[ \alpha_x \] \[ \Delta T \]
File Naming Conventions

- Perturbed input file name:
  \[ fn = \text{trim}(\text{ofile}(ic))//\text{trim}(\text{ext}) \]
  \[ l_1//"_"//\text{trim}(\text{parts}(ifl))//\text{trim}(\text{dimsc})//\text{pt}(ip1) \]

- `ofile` is for the case
  `data ofile/"c1", "c2", "c3"/

- `ext` is an “extension” for a version number

- `parts` is the part names
  `data parts/
  1 "Plate_","StudT_","StudB_","Spacer_"/

- `dimsc` is the perturbed dimension

- `pt` is m for negative perturbations and p for positive perturbations

- The code also writes the job submission scripts.

Los Alamos National Laboratory

Operated by Triad National Security, LLC for the U.S. Department of Energy's NNSA

UNCLASSIFIED
Going Further…

• Simplifying the model takes a long time, and I end up doing it more than once.

• So I use a program for that too:

c exclude(1:99,1:maxnucl)=0 ! default is keep all nuclides
c avgmatdens(1:ncases,1:99)=1.00
c dens_BR=1.2622d-01 ! Bottom Reflector unpert. density

c dens_SR=1.2632d-01 ! Side Reflector, Base unpert. density

c dens_IR(1:ncases)=0.40

c n simpl=22

c 1. Remove air.
c 2. Remove fill gas.
c 3. Center fuel, moderators, and absorbers.
c 4. Remove nylon screws and holes: preserve density.
c 5. Remove thermocouple probes, fill slots: preserve density.
c 6. Restore tray mass: combine trays and frames: use average density.

c if(n simpl.ge.6) then

c avgmatdens(1,23)=5.9692d-02

c avgmatdens(1,24)=avgmatdens(1,23)
c avgmatdens(2,23)=5.9711d-02

c avgmatdens(2,24)=avgmatdens(2,23)
c avgmatdens(3,23)=5.9798d-02

c avgmatdens(3,24)=avgmatdens(3,23)
c end if

c 7. Remove protruding part of RTD plugs: preserve density.
c 8. Remove slip nuts and compression studs, top, above compression spacer: preserve density.
c 9. Remove holes from tilt-weigh table and top plate: preserve density.
c 10. Remove knobs from compression studs, fill holes in platen adapter plate and core support plate: preserve density.
c 11. Eliminate hex nuts and washers.
c 12. Square rounded corners in upper adapter plate and top plate: preserve density.
c 13. Square rounded corners platen heat sink, bottom reflector, and side reflector, base: preserve density.
c 14. Restore mass of platen heat sink, bottom reflector, and side reflector, base.

c if(n simpl.ge.10) then

c avgmatdens(13,26)=5.9894d-02

c dens_BR=1.2619d-01

c dens_SR=1.2630d-01

c end if
Conclusions

- I recommend that we all stop cutting and pasting from spreadsheets.
  + Automate the things you do repeatedly.

- I believe that because of my system, the model is less likely to have errors than Section 3.
  + Our institutional bias against the model in favor of Section 3 is outdated.
  + For my evaluations, I give you permission to use the model instead of Section 3.

- How do *you* automate?
  + Talk to me; email me: favorite@lanl.gov

- Acknowledgment:
  
  This work was supported by the DOE Nuclear Criticality Safety Program, funded and managed by the National Nuclear Security Administration for the Department of Energy.